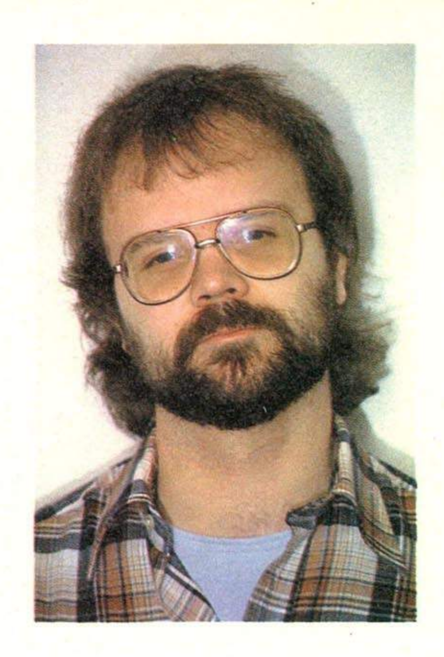

ATARI
ST
COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
Januar ´88
1
DM7,-
Ös 62,- Sfr 7,-
KLINGENDE CHIPS
Sound-Digitalisierer
FARB VIELFALT
Mit dem Omega-Grafik - System
SCHWARZ AUF WEISS
Der neue NEC-Multisync GS
KLEINE HELFER
Hard Disck Utilities auf dem Prüfstand
NEWS AUS USA
Comdex ´87



New Year, New Band, New Company

Halt, stop! Das war ja der Titel einer alten John Mayall Schallplatte, aber so ähnlich könnte es auch bei der ST Computer lauten: Neues Jahr, neue Leute in der Produktion (z.B. Layout) und eine in der Zwischenzeit nicht mehr ganz so neue "Merlin" Computer GmbH.

Ja, die ST Computer geht jetzt in das dritte Jahr. Als wir im Januar 1986 der staunenden Öffentlichkeit die erste Ausgabe präsentierten, wußte noch keiner, wie sie sich entwickeln würde. Wenn man die 64-seitige Premierenausgabe mit der Ausgabe vergleicht, die Sie jetzt in der Hand halten, fallen einem doch schon einige Unterschiede auf, die nicht zuletzt auch durch Sie, liebe Leser/innen, so deutlich ausgefallen sind. Sicher ist jedenfalls, daß es in Zukunft mehr Hardwareprojekte wie den Easytizer geben wird.

Damit die Qualität der ST Computer auch optisch besser wird, haben wir die ST Computer ab dieser Ausgabe in ein neues Gewand, sprich Layout, gesteckt. Dabei haben wir auch einigen Leservorschlägen Rechnung getragen. Deshalb – auch in Zukunft ist konstruktive Kritik jederzeit willkommen.

Bliebe zuletzt noch die "new company". Da die ST Computer ja eine Gemeinschaftsproduktion des Heim Verlags und der ehemaligen ST Computer Redaktion ist, aber letztere zusätzlich noch mit dem Vertrieb von Hardware (Easyprommer) und Software (PC ditto) zu tun hat, haben wir die ST Computer-Redaktion in eine GmbH umgewandelt, nämlich die "Merlin" Computer GmbH.

Auch im neuen Jahr werden wir weiterhin bemüht sein, die Qualität der ST Computer zu steigern. Doch dazu sind auch, wie bereits oben gesagt, Ihre Anregungen und Mitarbeit durch Beiträge nötig. Und seien Sie bitte nicht enttäuscht, wenn unser Telefon auch beim dritten Versuch besetzt ist, oder die Beantwortung Ihres Leserbriefes mal etwas länger dauert. Wir tun unser möglichstes, gelesen wird alles, aber wir haben nebenbei auch noch jeden Monat eine Zeitung mit einigem Umfang zu füllen. Deshalb versuchen wir im Moment ein Konzept zu entwikkeln, das auf Hotline – Basis arbeitet, das es uns erlauben soll, beides zu tun.

Ihr Harald Egel

Inhalt 1/88

Allgemeines

Software

Hardware

Grundlagen

Serie

Aktuelles

Klingende Chips – Sound-Digitalisierer im Vergleich

Mit den immer größer werdenden Speicherkapazitäten der Computer nutzen immer mehr Software-Hersteller, aber auch Hobbyprogrammierer, die Sound-Digitalisierung als einfaches und praktisches Mittel, die Mängel eines Soundchips zu umgehen. Doch selbst wenn der Chip brauchbar ist, ist es immer noch eine gehörige Portion Arbeit, ihn zu programmieren. Deshalb stellen wir Ihnen einige der auf dem Markt erhältlichen Geräte vor.



Programmierpraxis

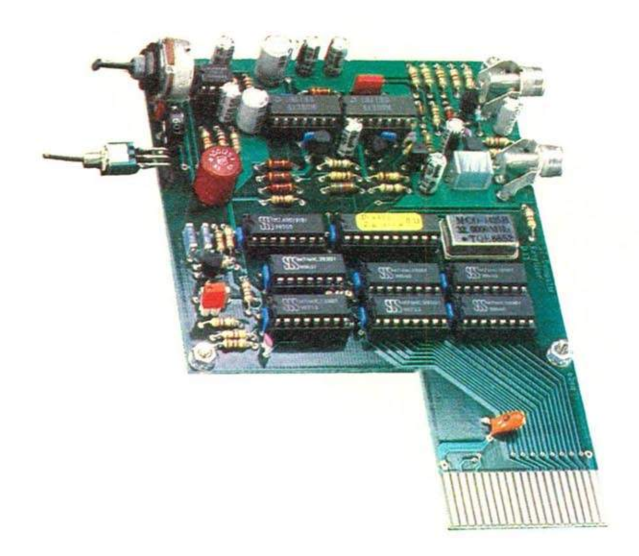
Machen Sie Ihren Computer zu einem Instrument, das Sie immer besser beherrschen lernen. Neue Tips und Tricks dazu geben wir in der dritten Folge unserer Programmierpraxis. Unter dem Motto "Programme von Lesern für Leser" ist sicher für jeden etwas dabei.

OMEGA Grafik-System

Die Firma OMEGA hat das erste Grafik-System für den ATARI ST entwickelt. Durch eine Zusatz-Grafik-Karte stehen bis zu 256 Farben gleichzeitig zur Verfügung, die dazu noch aus einer Palette von 256000 Farben ausgewählt werden können. Doch dies ist nur eine der vielen Möglichkeiten dieser Karte. Lesen Sie selbst.

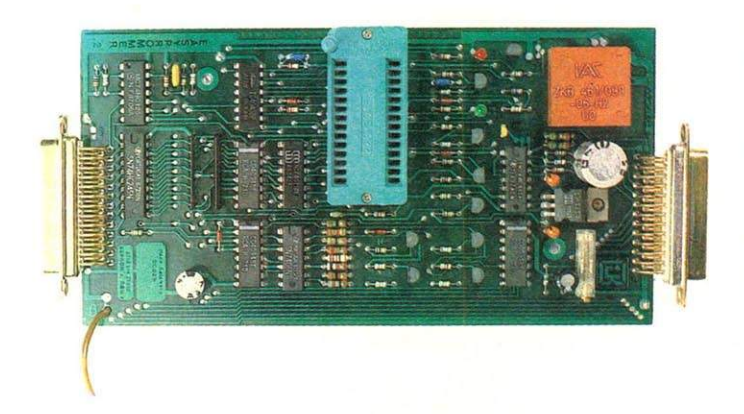
Aus der Hardwareküche





Der Easyprommer V2.2

Ein universelles Programmiergerät für den ATARI ST, das nicht nur alle gängigen EPROMs der 27...-Serie inkl. 27011 (Megabit-Eprom) brennt, sondern auch die modernen EEPROMs lesen und programmieren kann. Die sehr komfortable Software, natürlich voll GEM unterstützt, erlaubt alle nur denkbaren Manipulationen. Mit dem integrierten HEX/ASCII-Monitor lassen sich Änderungen der Daten blitzschnell realisieren. Der auf der Platine befindliche Druckertreiber erlaubt das problemlose Drucken der Daten. Zwei Zusatzprogramme ermöglichen das Brennen **jeder** nicht kopiergeschützten Software direkt von Diskette einschließlich Accessories und Autostart-Programmen. Der Easyprommer wird an den Druckerport angeschlossen und samt Diskette und Handbuch geliefert.



Easytizer
– der Videodigitizer ohne Geheimnisse

Mit dem Easytizer können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder auch direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen.

Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden **12,5 Bilder in der Sekunde** wiedergegeben! Im hochauflösenden Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben.

Besondere Merkmale des Easytizers:

- Software vollständig in Assembler
- Abspeichern der Bilder im DEGAS-Format
- Spiegeln eines Bildes in horizontaler und vertikaler Ebene
- Invertieren eines Bildes
- Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format verkleinert oder vergrößert werden
- Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder)
- Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible
- Wahlweise automatische oder manuelle Helligkeitseinstellung
- Schnappschuß

Lieferumfang:

1.) **Fertiggerät**
komplett aufgebaut und geprüft, inklusive Diskette mit der Easytizer Software und Bedienungsanleitung. Preis: DM 289,–

2.) **Teilesatz**
Für Bastler liefern wir einen Teilesatz bestehend aus:
- doppelseitiger, elektronisch geprüfter Platine mit Lötstopplack und Bestückungsaufdruck sowie vergoldeten Anschlußkontakten
- fertig programmiertes GAL 16V8
- Quarzoszillatormodul 32 MHz
- Diskette und Bedienungsanleitung — Preis: DM 129,–

Hiermit bestelle ich

☐ Easyprommer (Fertiggerät) für DM 349,–

☐ Easyprommer (Bausatz) für DM 298,–

☐ Easytizer (Fertiggerät) für DM 289,–

☐ Easytizer (Teilesatz) (w. o. beschrieben) für DM 129,–

☐ 128 KByte EPROM-Karte (fertig bestückt o. EPROMs) für DM 58,–

☐ Verlängerungskabel für Easyprommer für DM 39,90

Versandkosten: Inland DM 7,50 Ausland DM 10,–
Auslandsbestellungen **nur** gegen Vorauskasse

☐ Vorauskasse
☐ Nachnahme

Name: ____________ Vorname: ____________

Straße: ____________ Ort: ____________

Unterschrift: ____________

Bezugsquelle:



Industriestraße 26
6236 Eschborn

NEWS

Das erste Spiel von Application Systems

Nachdem die Heidelberger Softwareschmiede schon Anwenderprogramme wie z.B. Signum! oder STAD auf den Markt gebracht hat, kann man demnächst auch das erste Spiel erwerben. Bei BOLO handelt es sich um eine Breakout-Version, die ihresgleichen sucht. Bestechende Grafik und viele neue Features versprechen viel Spielfreude. Dabei heißt es, 50 Level zu überwinden, bis der Mega-Ghost geschlagen ist. Es wird sowohl ein Farb- als auch der Monochrommodus unterstützt. Geschrieben wurde es unter Megamax Modula 2, das auch noch im ersten Quartal '88 erhältlich sein wird. Der Preis wird unter DM 100,- liegen.

Bild 1: Durch Bolo wird die Breakout Spielidee vervollkomnet

Application Systems //Heidelberg
Englerstr. 3
6900 Heidelberg
Tel.: 06221/300002

Messen und Regeln mit dem ST

Die Firma WK-Elektronik bietet ein neu entwickeltes I/O-Interface-Bussystem für den ATARI ST an. Durch ein völlig neuartiges Konzept kann man den ST sehr einfach zum Messen, Regeln und Schalten einsetzen. Der flexible Aufbau ist für jedermann verständlich und bewirkt, daß sich das System nahezu beliebig erweitern läßt.

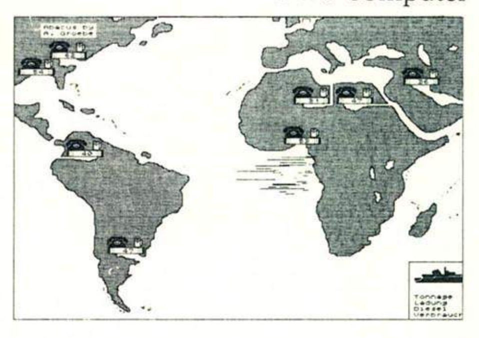


Bild 2: ABACUS, ein Wirtschaftsspiel

Das Grundsystem besteht aus einem Motherboard im Europakartenformat mit 6 Slots, wobei sich jeder Slot unabhängig adressieren läßt. Es lassen sich bis zu 16 Kartentypen gleichzeitig bedienen. An Baugruppenkarten steht folgendes zur Verfügung: 8-Kanal-digitale Eingabekarte, 8-Kanal-elektromechanische Ausgabekarte, 8-Kanal-elektronische Ausgabekarte, 8-Bit-Analog-/Digitalwandlerkarte, 8/16-Kanal-Analog-Multiplexerkarte und eine Experimentierplatine.

Voraussichtlicher Lieferbeginn ist Januar '88. Der Preis für ein System, bestehend aus Motherboard, Relais- und Eingabekarte, wird wahrscheinlich unter DM 200,- liegen.

WK-Elektronik
Postfach 4070
6911 Malsch
Tel.: 07253/23904

ABACUS, das Wirtschaftsspiel

Daß Computerspiele nicht nur krachen und knallen müssen, will die Firma BWB-Computer mit Ihrem neuen Wirtschaftsspiel ABACUS beweisen. Bei ABACUS können bis zu fünf Spieler aktiv über ihr eigenes Schicksal entscheiden. Hier darf man wie an der Börse spekulieren, Geschäfte tätigen, mit Devisen handeln, Aktien an- und verkaufen, sowie Intrigen schmieden. ABACUS kennt weder Grenzen im Einfallsreichtum noch in der Vielfalt unentdeckter Fähigkeiten. Es ist zum Subskriptionspreis von DM 59,- erhältlich.

BWB-Computer
Gleiwitzerstr. 1
6233 Kelkheim
Tel.: 06195/73346

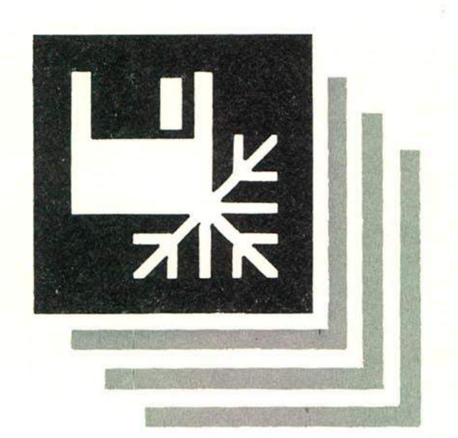
UPC / EAN-Barcodes mit dem ST

Für Händler, die kleinere Mengen an Produkten mit maschinenlesbaren Etiketten versehen wollen (z.B. Nachetikettierung), und bei denen Flexibilität groß geschrieben wird, gibt es jetzt ein Programm, mit dem aufklebbare Barcode-Symbole mit Matrixdruckern erstellt werden können. Es stehen zwei Versionen (9- oder 24-Nadel-Drucker) zur Verfügung.

Das Programm wird über Menüs weitestgehend mit der Maus gesteuert. Dort, wo es zweckmäßig erschien, wird die Bedienung von erläuternden Grafiken auf dem Monitor begleitet. Die Layout-Gestaltung bietet weitreichende Wahlmöglichkeiten an. Nebenher stellt es dem interessierten Laien eine Einführung in das Verschlüsselungsprinzip zur Verfügung. Der Preis beträgt für die Programmdiskette DM 135,–.

Hans-D. Groppler
Auf den Erlen 75
6200 Wiesbaden
Tel.: 06127/61357

SOFTWARE NEWS

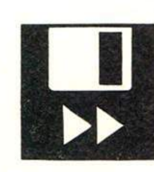


I ST Freezer™
· NEU ! Ermöglicht komprimierte Speicherdumps auf Disk · STOP/GO · Trainerversion für Spiele möglich · läuft auf S/W und Farbe sowie beiden TOS-Versionen mit 1 MB RAM · Beachten Sie bitte die (C)-Bestimmungen ! · Wird mit deutscher Anleitung geliefert.
Preis: DM 148,-*

LISP-Complet™

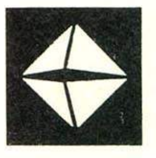


· Superangebot · kompletter LISP-Lehrgang und GEM-Programmiersystem · zwei Handbücher für Lehrgang und Programmiersystem (deutsch) · kein Kopierschutz · läuft auf S/W und Farbe auf allen TOS-Versionen
Preis: DM 98,-*



I ST Speeder™
· Beschleunigt das Arbeiten mit Diskette bis zu Faktor 10 !· Arbeitet mit intelligentem Cache-Memory · Ein wirkliches Muß für den professionellen Anwender · Arbeitet mit neuem und altem ROM, S/W und Farbe · Wird mit deutscher Anleitung geliefert
Preis: DM 89,-*

TrashHeap™



· Das ultimative 3D-Weltraumspiel · Wurde schon im Fernsehen und Radio vorgestellt · Wird mit 3D-Brille und Anleitung (deutsch) geliefert · läuft auf S/W und Farbe sowie unter beiden TOS-Versionen · Digisound
Preis: DM 69,-*

BTX ST/ALADIN™



· Endlich BTX auf dem ST ! · Telex über BTX · Seitenabrufe vorprogrammierbar, autom. Einloggen · Einf. von Grafik/Texten u.v.m. · Erf. ALADIN (Soft-/Hardware) und S/W-Monitor · Upgrades für TOS lieferbar etwa 2/88 · selbstdef. Zeichen
Preis: DM 398,-*

Programmierer für ATARI ST/AMIGA gesucht – Schreiben Sie uns !



Intelligent Spooler ST™
· Intelligenter Drucker-Spooler · verbraucht nur 50 KB · Reihenfolge der Texte änderbar · Speichert bis zu 16 MB-Daten · Hardcopy spoolbar · läuft auf S/W und Farbe
Preis: DM 98,-*

Dizzy Wizard™



· Das Spiel, das keinen Blitter braucht · 100 verschiedene Levels · bis zu drei Spieler · läuft auf S/W und Farbe sowie neuem/altem TOS · Test in ST 9/87 · Digisound
Preis: DM 69,-*

Crypt_it™ – Daten und Programmverschlüsselung DM 98,-*
Lock_it™ – Der Kopierschutz für Anspruchsvolle DM 298,-*
BinLook™ – Die Bildergalerie unter DEGAS für den ATARI ST DM 79,-*
MusiX32™ – Der Musikstandard auf dem ST DM 89,-*
Laser Deluxe™- P 6 und FX 80-Emulator für ATARI-Laser DM 248,-*

Alle Programme werden auf einseitigen Disketten ausgeliefert !

ANRUF GENÜGT: 069 / 61 40 46

Bestellcoupon

☐ Bitte senden Sie mir kostenloses Informationsmaterial (DM 1,40 Rückporto liegt bei)
☐ Hiermit bestelle ich

____

zzgl. DM 6,- Versandkosten (bei Vorauskasse)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei
Name ____________________
Straße ____________________
Ort ____________________ ST 1/88

* Alle Preise sind empfohlene Verkaufspreise ! Händleranfragen erwünscht !

TOMMY SOFTWARE®

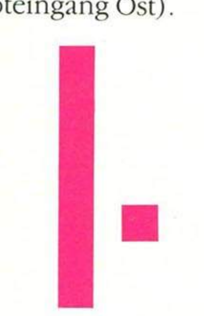
Gutzkowstr. 35, D-6000 Frankfurt/M 70, Tel./BTX: 069 / 61 40 46, Telex über BTX, KEIN LADENVERKAUF

Schweiz: Senn Computer AG, Langstr. 31, CH-8021 Zürich, Tel.: 01 / 24 17 37 3 Österreich: Computer-Studio Wehsner GmbH, Paniglgasse 18-20, A-1040 Wien, Tel.: 0222 / 65 78 08

Irrtum und Änderungen vorbehalten.

Amateurfunk auf der Chip Hamburg '88

Erstmalig wird vom 21. – 24. Januar 1988 auf dem Hamburger Messegelände in der neuen Halle 12 eine große Elektronik- und Computermesse unter der Bezeichnung "Chip '88" veranstaltet werden, auf der u.a. Hobby-Elektronik, Mikrocomputer, Software, Computer-Grafik, Modellfernsteuerung, HiFi-Technik und Seatronik gezeigt werden. Auch die Hamburger Funkamateure werden mit einem Gemeinschaftsstand des DARC-Bezirksverbands vertreten sein und modernste Kommunikationstechniken des Amateurfunks wie Packet-Radio, RTTY usw. vorführen. Es werden allgemeine Informationen erteilt. Die "Chip '88" ist an allen Tagen von 10 – 18 Uhr geöffnet (Haupteingang Ost).



COMDEX FALL '87

Verkehrte Welt: Die Computer regieren Las Vegas, die Glücksspieler Wallstreet

Es war die Zeit, zu der Roulette im Vergleich zu Aktienmarkt und Dollarkurs als krisensichere Kapitalanlage gehandelt wurde. Wallstreet war dabei, Las Vegas den Rang als größtes Spielcasino der Welt mühelos abzulaufen.

"...die Welt wartet auf ein Wort des Präsidenten... (Time Magazin vom 5. November).

Doch wenn es wirklich so einfach wäre, könnte jeder Melonenfarmer oder mittelmäßige Opernsänger Präsident der Vereinigten Staaten werden.

Ronnie jedenfalls hat es verstanden, Prioritäten zu setzen. Nach dem Motto: "Wo die Computer sind, ist die Macht.", hat er als erster in der "Computer Show Daily" vom 3. November die COMDEX-Besucher persönlich begrüßt. Wallstreet und der Rest der Welt mußten noch ein paar Tage warten.

Die Comdex fand vom 1. bis 6. November in Las Vegas statt. Es handelte sich bei dieser Messe um eine reine PC-Messe, auf der sich fast alle Firmen, die in diesem Bereich tätig sind, ein Stelldichein gaben. Am häufigsten waren demnach natürlich PCs, ATs und 80386er zu sehen. Den Vogel schoß eine 80386-Workstation mit 25 (in Worten fünfundzwanzig) MHz Taktfrequenz ab. Die Arbeitsgeschwindigkeit soll bei der eines ATs mit 32.5 MHz liegen. Man wurde von einer Flut von billigen Motherboards für 80386-Maschinen fast überschwemmt. Ähnlich unüberschaubar war das Angebot von Erweiterungskarten für PC/AT-kompatible Rechner.

Auch Atari ist gekommen, mit Sack und Pack und des Königs neuen Kleidern. Die neuen königlichen "Herrenausstatter" des Hauses Atari sind die britische Entwicklungsfirma Perhellion und der Chiphersteller Inmos. Diese beiden Firmen haben es Atari ermöglicht, der staunenden Welt ABAQ, einen Supermicrocomputer, vorzustellen, der auf dem Transputer-Chip von Inmos basiert. Der Prototyp, den Atari in Las Vegas vorstellte, ist wohl eher ein Dornröschen als ein Erlkönig, aber bis zur offiziellen EuropaPräsentation in Hannover wollen ihm Shiraz M. Shivji und sein Team noch eine ganze Menge Leben einhauchen. Die ersten Exemplare sollen dann Ende 1988 über den Ladentisch gezogen werden.

Das Herz des neuen Rechners ist der T800 mit einer Taktfrequenz von 20 MHz. Der Chip leistet dabei 10 MIPS (**M**illionen **I**nstruktionen **P**ro **S**ekunde) und 1.5 MFLOP (**M**illionen **FL**ießkomma **OP**erationen pro Sekunde). Es können 3 gepufferte Transputerlinks, die ebenfals mit 20 MHz getaktet sind, verwendet werden. Die Grundausstattung soll eine Speicherkapazität von 4 Megabyte RAM und einem Megabyte Bildschirmspeicher (VRAM) haben. Der Bildschirm kann in 4 Modis betrieben werden:

1280 * 960 4 bits/Pixel
1024 * 768 8 bits/Pixel
640 * 480 8 bits/Pixel
(2 Bildschirmseiten)
512 * 480 32 bits/Pixel

Auf der Transputerplatine soll ein Colorblitter seinen Dienst verrichten, um einen schnellen Bildschirmaufbau zu gewährleisten. Was Schnittstellen angeht, setzt ATARI bei dem neuen Transputerrechner auf Standards. Es wird eine DMA-fähige SCSI Schnittstelle geben, für die eine 40 MB-Winchester geplant ist. Auf der Hauptplatine sind 3 Slots für Erweiterungskarten vorgesehen. Sie ermöglichen es, die Speicherkapazität bis auf 64 MB auszubauen, oder das System auf maximal 13 Transputer aufzurüsten. Der gute alte ST wird am Transputerrechner nur ein karges Dasein fristen. Er wird zum Ein-/Ausgabe-Prozessor für Tastatur, Maus, Floppy und was es sonst noch so an Peripherie gibt, degradiert. Hinter vorgehaltener Hand ließ man verlauten, daß er selbst dafür bald überflüssig würde. Kein Wunder, denn wozu braucht man 4 MB Tastaturpuffer, Floppycache oder Druckerpuffer?

Das Betriebssystem für die neue Maschine trägt den Namen Helios und soll UNIX-kompatibel sein. C-Programme, die als Quellcode vorhanden sind, sollen mit minimalen Änderungen auch unter Helios compilierbar und lauffähig sein. Wer schon Erfahrung mit den verschiedenen UNIX-Versionen hat, kann den Wahrheitsgehalt solcher Versprechen beurteilen. Aber warten wir ab. Als Benutzeroberfläche für Helios soll es einen Command-line-interpreter geben, der mit der C-Shell vergleichbar ist und zusätzlich X-Windows Version 11, ein

Windowsystem aus der UNIX-Welt, hat. Das Design von Helios ist auf Parallelverarbeitung ausgelegt. So ist es möglich, mehrere Prozesse auf verschiedenen Transputern gleichzeitig laufen zu lassen. Die Hardwarestruktur der Transputer mit den Links ist besonders geeignet, Prozesse, die durch eine Pipe verbunden sind, auf verschiedenen Transputern laufen zu lassen, und die Daten via Link zu übertragen. Dieser Mechanismus wird von Helios unterstützt, und es fällt nicht schwer, sich vorzustellen, wie hoch dadurch die Arbeitsgeschwindigkeit wird.

Der nächste größere Wurf von Atari wird dann wohl die 80386-Maschine sein. Die erste öffentliche Vorstellung dieses Gerätes ist angekündigt für die CEBIT 88, die Auslieferung soll im dritten Quartal des Jahres beginnen. Was allerdigs an dem von Atari ausgestreuten Gerücht der "speed processor option" von 16 MHz und 20 MHz dran ist, muß sich erst noch zeigen.

Ebenfalls von Atari vorgestellt wurde eine neue Serie von Hard-Disk-Drives genannt, MEGA FILE. MEGA FILE 20 (20 Megabyte) ist das Grundgerät der Serie. Es hat es einen 19-Pin-DMA-Anschluß für daisy-chaining. Zur Zeit gibt es noch MEGA FILE 40. Wann weitere Geräte aus dieser Serie erscheinen, steht jetzt noch nicht fest.

Besonders stolz war man auch auf den präsentierten CD-ROM/CD-AUDIO Player. Die optische Scheibe kann jetzt nicht mehr nur Musik wiedergeben, sondern auch 540 Megabyte Daten verwalten. Atari hat auch die Audio Playback Option eingebaut, so daß das CD-ROM auch die Fähigkeit besitzt, gewöhnliche AUDIO COMPACT DISKS in hervorragender Qualität wiederzugeben. Die Kapazität von 540 Megabytes entspricht der Datenmenge, die man auf über 1000 normalen Disketten oder ca. 200.000 Druckseiten unterbringen kann. Wie gut dieses Gerät wirklich ist, werden wir Ihnen verraten, wenn wir die erste CD-Sammlung damit kopiert haben.

Mehr oder weniger gut kopiert für den US-Markt werden zur Zeit für den Atari ST diverse aus Deutschland kommende Softwareentwicklungen. Manchmal werden sie auch einfach importiert. So war die Beteiligung an ausstellenden Firmen für den ST zwar wirklich beachtlich, aber die Ausbeute für den gutinformierten ST-Computer-Leser ist nicht unbedingt schränkefüllend. Interessanterweise wird, genau wie in Großbritanien, der ST eher als Spielerechner eingestuft. Renommierte Firmen, wie z.B. HaBa haben die Konsequenzen daraus gezogen und sich von diesem Markt verabschiedet.

Auf der anderen Seite gibt es immer wieder Firmen, die ganz neu in diesen Markt einsteigen oder ihre Palette um Atari ST Produkte erweitern.

Eine davon ist Foresight Resources Corporation. Sie hat die bekannten CAD-Programme der DRAFIX-Serie für den IBM entwickelt. Die Umsetzung von DRAFIX 1 für den Atari ST wurde jetzt auf der Comdex in Las Vegas präsentiert. Das "factsheet" verspricht einfache Handhabung, sehr viele hilfreiche Befehle wie z.B. stretch, shorten, trim, bevel, round, fillet, erase, unerase etc., 16 Zeichenstifte, 11 verschiedene Fonts u.v.a.

Welchen Platz im Konzert der zahlreichen CAD-Programme Drafix 1 beim Atari ST einnehmen wird, muß sich aber noch zeigen. Ebenso erweisen wird sich der Stellenwert von IDVIEW. Es handelt sich um ein dreidimensionales CAD (computer added design)-Paket für den Mega ST. Die "features", die IntelliDesign dabei anpreist, sind ähnlich den oben beschriebenen, ergo: test and see.

Ebenfalls jede Menge neuer "features" – um genau zu sein, über hundert – werden von SoftLogik für die "professional" -Version des bekannten Desktop-Programms PUBLISHING PARTNER angekündigt. Gleichzeitig wurde der Preis der Original-Version (schönes Wort für veraltet) von 149 US$ auf 89 US$ gesenkt. PUBLISHING PARTNER PROFESSIONAL wird jetzt für 149 US$ angeboten. Die neuen Fähigkeiten sind u.a. automatisches Fließen des Textes um jede Graphik, das Einbinden von Degas Compressed-, Easy Draw- oder IMG-Bildern, Zoom, sowie das Importieren von Word Perfect-, First Word- und Word Writer-Files.

Eine Möglichkeit, die dazu passenden Bilder selbst zu zeichnen, bietet ProTablet ST von Quantum Microsystems Inc. Laut QMI soll das "digitizer tablet" mit einem Verkaufspreis von 395 US$ besonders günstig werden. Es hat eine Arbeitsfläche von ca. 30 x 20 cm (12 x 8 Zoll) und Gesamtmaße von 45 x 28 x 3 cm. Die Arbeitsgeschwindigkeit von 60 Pixel pro Sekunde und die Auflösung von 100 Zeilen pro Zentimeter werden auch bei der Anfang 1988 erscheinenden größeren Version 13.0 x 16.5 Zoll (33 x 42 cm) gleich bleiben. Sie soll dann 595 US$ kosten. Ein Treiber-Programm sowie benötigte Kabel und Hardware sind jeweils inbegriffen. Ob und wie sich ProTablet ST auf dem deutschen Markt mit vergleichbaren Produkten auseinandersetzten wird, bleibt abzuwarten.

Ebenfalls ungeklärt ist noch die Frage des deutschen Distributors für DO-IT und DESK CART. Der Slogan "DOS Instead of TOS" beschreibt die Zielrichtung von DO-IT. Ein "complete command line interpreter" ersetzt das Desk Top duch eine MS-DOS-ähnliche Umgebung. Alle unter GEM oder GEM applications beinhalteten Programme laufen problemlos. Die über 50 "internal commands" wie black, dstat, kbrate, path und wrap sind im Begleitheft ausführlich erklärt. Leider liegt bisher aber noch keine deutsche Übersetzung vor. Vorliegen, neben DO-IT, tut uns aber auch DESK CART. Es ist ein Desk Accessory & Clock Cartridge. Nach dem Einstecken in den ST hat der Benutzer Zugriff auf Kalender, Termine, Notizbuch, Taschenrechner, Schreibprogramm, Adreßbuch, Telefonliste VT-52 Terminal, Disk Utilities u.a.m.. Das Cartridge hat eigene 64k-ROM und arbeitet problemlos mit Gem-Applikationen zusammen. Der amerikanische Preis beträgt 99.95 US$ (DO-IT 24.95 US$).

Nochmals überarbeitet hat Jefferson Software das MODULA-2 DEVELOPMENT SYSTEM. Das Programm arbeitet mit der dritten Ausgabe von Modula. Dies ist die neueste von Dr. Niklaus Wirth (auch Entwickler

von Pascal) entwickelte Version. Updates für Besitzer von Jefferson Modula 2 V1.0 sind kostenlos. MODULA 2 V1.1 wird in zwei "Preisklassen" angeboten: Das "normale" Paket kostet 49.95 US$, das DEVELOPMENT SYSTEM 79.95 US$. Die Entwickler-Ausgabe bietet zusätzlich eine ausführliche gedruckte Dokumentation, mehr "libraries" und mehr Programmbeispiele. Beide Versionen werden inklusive Source-Codes geliefert.

Fertig und in Vegas erstmals vorgestellt worden, ist hingegen der neue FORTRAN COMPILER von Prospero. Er liegt bereits in der Redaktion vor. Das Paket ist nur in englischer Sprache erhältlich. Deshalb begnüge ich mich hier mit einem kurzen Auszug aus dem "factsheet":

1) Vier-Fenster-Source-Editor
2) 289 seitiges Handbuch zur Nutzung von AES Bindings
3) 247 seitiges Handbuch zur Nutzung von VDI Bindings
4) Mächtiger symbolischer Debugger
5) Verbesserter Linker
6) Verbesserte Kompatibilität des Compilers

Beim LATTICE C COMPILER müssen Sie noch bis Hannover mit Ihrer derzeitigen Version klarkommen. Falls alles klappt soll dort die Version 4.0 vorgestellt werden. Der LATTICE C COMPILER ist in der Bundesrepublik bei zahlreichen Händlern erhältlich (vgl. bereits erschienene ST COMPUTER-Ausgaben).

Bereits bei anderen Messen vorgestellt wurden auch die meisten der in Las Vegas präsentierten Spiele. Wir können uns also hier auf die wenigen wirklichen Neuvorstellungen konzentrieren.

Jetzt endlich erschienen ist DEFENDER OF THE CROWN von Cinemaware, eine Mischung aus Strategie und Action. Es spielt im mittelalterlichen England. Die Atari ST Ausgabe ist bei weitem die spielstärkste Version, die bisher vorgestellt wurde und hat viele zusätzliche Features gegenüber Versionen auf anderen Rechnern.

Ebenfalls so gut wie fertig, aber bei der Messe noch nicht erhältlich, war KING OF CHICAGO. Es spielt in der Zeit Al Capones und dreht sich um die Bandenkriege der dreißiger Jahre.

In den USA vorgestellt werden die Cinemawarespiele von Mindscape. Ebenso von Mindscape repräsentiert wird Hewson Consultants. Ehemalige C64-Besitzer ahnen was jetzt kommt – es stimmt. DAS Weltraumschießspiel wird demnächst für den Atari ST erscheinen – seine Idee wurde oft kopiert, aber das Original wurde nie übertroffen – URIDIUM. Der Vertrieb in Deutschland ist noch nicht geklärt. Wir werden aber so bald wie möglich mit mehr Informationen aufwarten.

Epyx war da, aber über alle Titel, die vorgestellt wurden, haben wir bereits im Messebericht von des Sommer-CES berichtet. Was wann wirklich erscheint – ...???

Electronic Arts hat sich – Erscheinungstermine betreffend – diesmal selbst in die Pflicht genommen. Fest zugesagt für November ist GONE FISH'N von Interstel. Es basiert auf dem Konflikt des passionierten Anglers, der eine Entscheidung treffen muß zwischen arbeiten und angeln gehen. Sie haben einen eigenen kleinen Laden, der Ihnen Ihr Hobby erst ermöglicht. Ziel ist es, durch geschickte Planung das Geschäft erfolgreich zu führen, und trotzdem möglichst viel Zeit am See zu verbringen. Nicht ganz so beschaulich geht es bei STARFLEET II – KRELLAN COMMANDER zu. Die Fortsetzung der Starfleet Saga soll im Dezember erscheinen und über 35 galaktische Regionen mit mehr als 1700 Sternensystemen Ihrem Eroberungsdrang preisgeben. Vor die Herrschaft über das All hat Interstel aber zahlreiche Raumschlachten und durchkämpfte Nächte gesetzt.

Ein Bonbon am Rande war der von CSS angebotene "Winterpelz" für die Maus. Diese und andere Spielereien sind im kostenlosen Katalog von Computer Software Service.

Wie die meisten von Ihnen wissen, gibt es für den ST eine Reihe von Programmen, die einen anderen Rechner simulieren. Die interessanteste Neuheit auf diesem Gebiet stellt Magig Sac+ dar. Der bereits in Heft 2/87 der ST-Computer getestete Macintosh-Emulator wurde um wesentliche Features erweitert. Es ist in der neuen Version möglich, die Harddisk des ST auch für den Macintosh-Betrieb zu nutzen, und auch von Harddisk zu booten. Der Speicherausbau der Mega-STs wird voll unterstützt, d.h., bis zu 3900 KByte können unter dem Mac-System verwendet werden. Der Clou ist jedoch eine Hardware-Erweiterung, die es ermöglicht, Mac-Disketten mit den Atari-Laufwerken zu lesen. Das Kästchen mit den Ausmaßen von 2 Zigarettenpäckchen wird einfach auf den Floppybus gesteckt, und man kann umschalten zwischen Macintosh- und Atariformat. Das mühevolle Übertragen von Dateien über die serielle Schnittstelle entfällt damit. Der Preis für den Emulator beträgt 280 Dollar. Leider war nicht in Erfahrung zu bringen, wann das tolle Kästchen in Deutschland verfügbar sein wird.

Mehrere interessante Produkte gab es an dem Stand von Supra Corporation zu sehen. Eine Harddisk mit 250 MB wurde angeboten und ist mit ca. 4000 Dollar nicht zu teuer, wenn man bedenkt was 12 SH 205 mit 20 MB kosten. Für die stolzen Besitzer von Mega ST's gab es einen besonderen Leckerbissen. Die Harddisk zum Einbau in das Gehäuse des Mega ST mit 20 MB für 650 Dollar. Oder für alle STs die 10 MB-Floppy mit 80 ms Zugriffszeit für 895 Dollar.

Eine Abschlußbemerkung zu der ganzen Messe: In mancher Hinsicht sitzen wir in Deutschland noch auf den Bäumen. Umgedreht sitzen viele Firmen in Amerika auf dem hohen Roß (Germany ? That's in Europe, isn't it ?). Auf jeden Fall weiß man nach dem Besuch einer amerikanischen Messe wie der COMDEX in der Regel mehr als die deutschen Niederlassungen der amerikanischen Firmen.

(CPL/JL)

Klingende Chips

Sound Sampler im Vergleich



Auf der Suche nach Techniken, die unseren geliebten Computer immer menschenähnlicher werden lassen, war die Digitalisierung von Tönen ein grosser Schritt. Die Zeiten der mühsamen Sprachsynthese, in der wir über eine kaum verständliche Metallstimme jubelten, sind in vielen Bereichen vergessen. Durch die kontinuierlich wachsenden Speichermengen und die Produktion schneller und billiger Analog-/Digital-Wandler stand der direkten Sprachaufzeichnung nichts mehr im Wege.

In diesem Artikel werden drei Soundsampler-Systeme vorgestellt. Unter einem System verstehen wir dabei Hardware und Software. Die Hardware ist nötig, da der ST nicht über einen eingebauten Wandler-Baustein verfügt, aber dazu mehr im Infokasten.

Natürlich gibt es noch wesentlich mehr Systeme, die sich in der Leistung und vor allem im Preis stark unterscheiden. Die Leistungen reichen dabei von der einfachen Unterhaltung bis hin zur semiprofessionellen Anwendung. Professionelle Geräte haben eines gemeinsam: den hohen Preis, weshalb sie auch in diesem Vergleich keinen Platz finden.

Der Bunte

PRO SOUND DESIGNER VON EIDERSOFT

Dieses System macht auf den ersten Blick den besten Eindruck. Es besticht durch eine eindrucksvolle Verpackung und eine übersichtliche Anleitung mit korrekter Übersetzung aus dem Englischen (dies ist nicht unbedingt üblich). Das eigentliche, sehr handliche Gerät wird direkt an den Druckerport angeschlossen. Zum Betrieb benötigt es zusätzlich eine 9 Volt-Batterie.

Color only

Spätestens beim Laden des Programms erwartet Sie (vielleicht) die erste Enttäuschung: Was weder auf der Verpackung noch auf den ersten Seiten des Handbuches aufgefallen war, wird plötzlich sichtbar: Das Programm läuft nur auf einem Farbmonitor!!! Das Weiterblättern im Handbuch kann das nur bestätigen:
Besitzer eines Monochrommonitors müssen auf einen schon überfälligen Termin für eine Anpassung warten. Nach Auskunft des deutschen Distributors 'CCD', wird das Programm derzeit völlig umgearbeitet. Zu diesem Zeitpunkt war es jedoch noch nicht fertig.

Nach dem Laden erscheint ein Kontrollbildschirm auf dem alle Funktionen über die Maus abrufbar sind: Es bestehen gewisse Ähnlichkeiten zu einer Videorekordertastatur. Weiterhin existieren noch eine Reihe Ikons und Funktionstasten-Symbole.

Aufnahme

Die beste Qualität erreicht

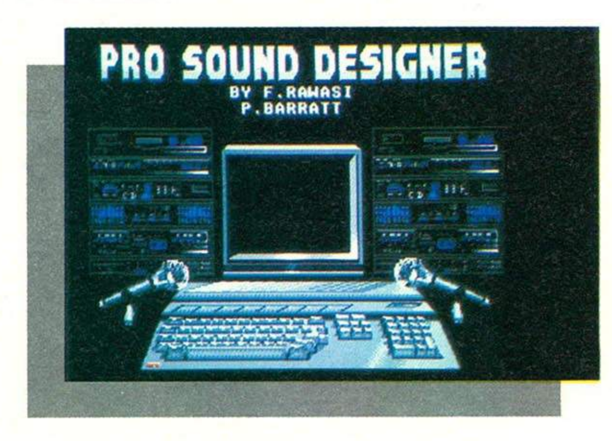


Bild 1: Das Einschaltbild des Pro Sound Designer

man, wenn man das Gerät mit seinem 3,5mm-Stecker mit dem Kopfhörerausgang der Hi-Fi-Anlage verbindet. Das passende Kabel dürfte jeder Radiohändler führen, es kann aber auch leicht selbstgebaut werden.

Sobald am Eingang ein Signal anliegt, beginnt im oberen Fenster eine Kurve zu tanzen. Dadurch hat man eine Kontrolle über die Aussteuerung des Signals und kann diese am Lautstärkeregler nachstellen. Dies ist äußerst wichtig, denn ein übersteuertes Signal klingt einfach grausig.

Weiterhin kann man die Abtastgeschwindigkeit im Bereich zwischen 3 und 30 KHz festlegen, wobei eine Geschwindigkeit von 3 Khz allerdings sogar zur Sprachaufnahme ungeeignet ist.

Die Aufnahme erfolgt solange, bis der Speicher voll ist oder die Leertaste betätigt wird. Das gesamte Sample erscheint grafisch dargestellt als langes Linienband.

Bearbeitung

Dazu schneidet man zuerst mit zwei Marken das zu bearbeitende Feld aus. Danach kann man es dann löschen, wiederholen, überspielen (Echoeffekte), drehen, neuaufnehmen (Sequenz bauen) oder vergrössern (um genauer abzuschneiden). Leider steht, was besonders bei der Löschfunktion negativ in Erscheinung tritt, keine Undo-Funktion zur Verfügung, um irrtümliche Eingaben rückgängig zu machen.

Wiedergabe

Wenn sie nicht die richtige Version mit dem sogenannten "Hi-Fi Upgrade Kit" besitzen, können Sie nur Ihren Monitor als Ausgabegerät benutzen und verpassen somit "das beste Ergebnis, aber setzen Sie sich in Verbindung mit uns, usw...". Warum diese Ausgabemöglichkeit von der Herstellerfirma nur als Zusatzteil betrachtet wurde, ist ein Rätsel. Bei der hier vorgestellten Version ist es jedoch enthalten und im Preis inbegriffen.

Neben der einfachen Tonwiedergabe des sich im Speicher befindlichen Samples besteht die Möglichkeit die Funktionstasten mit einzelnen Sequenzen zu belegen; eine Art Minitastatur mit 8 Tasten.

Extras

Das Programm sieht es vor, die Wiedergabe durch ein externes, akkustisches Signal auszulösen; der Ansprechpegel ist dabei variabel. Vielleicht kann man damit die Schwiegermutter erschrecken.

Die wohl interessanteste Möglichkeit ist das Einbauen von gesampleten Sequenzen in eigene Programme. Im Handbuch findet man dazu Erläuterungen und auf der Programmdiskette die nötigen Maschinencoderoutinen.

Resumee

Es ist zu bedauern, daß sich die mit der Verpackung erstrebte Professionalität nicht ganz auf die Software übertrug. Befehle zum Ändern der Farben sind ebenso unnötig wie das Bremsen der Ablaufgeschwindigkeit bei der Mausbewegung. Ein interessantes Plus ist allerdings der eingebaute Soundchip-Editor, mit dem man die verschiedenen Register des AY-3-89190 ansprechen und ihm dadurch alle möglichen Geräusche entlocken kann. Wer diese mit gesampletem Sound kombinieren möchte und mit einem Farbmonitor arbeitet, dem sei zu diesem Gerät geraten.

Der Sparsame

ST REPLAY VON MICRODEAL

ST-REPLAY beinhaltet ein ähnlich großes Hardwaremodul wie das vorherige Modell. Im Handbuch liegt allerdings eine deutliche Schwäche. Der Herstellerfirma Microdeal hängt leider der Ruf an, recht seltsame Übersetzungen zu liefern, die nicht selten zum Original kontroverse Beschreibungen liefern. Als extremes Beispiel wurde das Hardware-Modul mit "Diskette" übersetzt, die man dann in den Cartridgeport stecken sollte. Wer sich an diese Anleitung hält, wird sich sicherlich wundern. Der ebenfalls mitgelieferte englische Text erlaubt aber eine persönliche Interpretation des Handbuches. Trotz allem bleibt positiv zu bemerken, daß überhaupt eine deutsche Anleitung beiliegt – nicht jedes englischsprachige Softwareprodukt hat das zu bieten.

Das Modul wird entgegen den anderen zwei Produkten an den Cardridgeport angeschlossen. Dadurch entfällt die externe Stromversorgung per Batterie oder Netzteil. Der Druckerport bleibt dadurch dem Drukker vorbehalten, der ROM-Port ist natürlich belegt, aber irgendwo muß man ja an den Rechner ran. Die Verbindung mit der Tonquelle erfolgt über gängige ChinchBuchsen und belegt lediglich zwei Buchsen an der Stereo-Anlage.

Die Software

Nach dem Laden erscheint ein Kontrollbildschirm, der durch seine Einfachheit beeindruckt.

Bild 2: Das Menü des Pro Sound Designers



Der Versuch Funktionen mit der Maus anzusprechen, scheitert, denn die Befehle sind nur über Funktionstasten oder Tastenkombinationen auszuführen. Eigentlich schade, denn wer sich an den grauen Helfer gewöhnt hat, wird ihn sicherlich vermissen.

Laut Handbuch wurde der Be-

dienungskomfort nur gesenkt, damit der maximale Speicherplatz zur Verfügung steht. Zwar geht ST-REPLAY wirklich am sparsamsten mit dem Speicher um, jedoch bleibt fraglich, ob mit diesem Konzept der richtige Weg eingeschlagen wurde.

Manche Befehle wurden auf der Tastatur so dicht nebeneinander gelegt, daß die Gefahr, irreparable Eingaben zu machen, recht groß ist. So liegt der Befehl zum Sample löschen (ohne Bestätigung!!) direkt neben der Lautstärkeeinstellung.

Aufnahme

Nachdem man das Wandlermodul an den Kopfhörerausgang seiner HiFiAnlage angeschlossen hat (Kabel bitte selbst besorgen), empfiehlt es sich die Lautstärke anzupassen. Das geschieht mittels einer auf dem Bildschirm erscheinenden Aussteuerungsanzeige.

Die zweite, auch etwas direktere Methode, ist das 'Monitoring', welches das direkte Anhören der Aufnahmequelle nach dem Wandeln erlaubt. Dadurch lassen sich Übersteuerungen und auch klangliche Unterschiede in Abhängigkeit von der Samplingrate erkennen.

Ist die Qualität nicht befriedigend, kann man durch sofortiges Probieren die Samplefrequenz erhöhen, und zwar, bis sie ausreichend ist. Bei einfacher Sprache genügt eine Samplingrate von 10 Khz. Musik hingegen erfordert die höchste Qualität. Insgesamt stehen sechs verschiedene Frequenzen zwischen 5 und 31 KHz zur Verfügung.

Nach erfolgter Einstellung beginnt die Aufnahme, und so lange, bis der Speicher voll ist, oder sich Ihr Finger auf der Escape-Taste niederläßt. Die Aufnahme wird dann als Kurve im Editorfenster dargestellt.

Klangbearbeitung

Ähnlich wie beim Prosound erfolgt auch die Bearbeitung des Sounds. Der mit zwei Marken begrenzbare Ausschnitt steht zum Hören, Löschen, Verbinden, Wiederholen, Invertieren, Neuaufnehmen, Laden oder Speichern bereit. Eine zusätzliche Funktion "Fading", ermöglicht das Ausblenden am Ende des Stücks und verhindert so manche abrupte Endung.

Wiedergabe

Die Alternate-Taste spielt hierbei eine große Rolle, denn sie schaltet zwischen Monitorausgabe und direkter Ausgabe über den Wandler auf einen Verstärker um. Das gespeicherte Sample kann nun entweder in voller Länge oder nur ausschnittsweise ausgegeben werden. Zum Einbinden in eigene Programme stehen auf Diskette die nötigen Routinen zur Verfügung.

Eine andere Möglichkeit ist die Verwendung der von ST-Replay aufgenommenen Samples in das ebenso aus dem Hause Microdeal stammende DIGI-DRUM. Es handelt sich dabei um eine Rhythmusmaschine, die in bestimmten Rythmen die gesampleten Sounds ablaufen läßt. Damit können Sie eigene Geräusche, z.B. das Bellen Ihres Hundes oder das Startgeräusch Ihres Porsche (natürlich das des Zweitwagens, der Jaguar ist dazu zu leise!) in den Rhythmus einbauen.

Kurz gefaßt, ist ST-REPLAY ein einfaches System, das durch sein Konzept (DIGI-DRUM und Einbau in eigene Programme) gute Zwecke erfüllt. Aufgrund seiner sparsamen Software bietet es die längste Samplezeit aller Systeme, beherrscht allerdings nur die wichtigsten Befehle.

Der Alleskönner

AS SOUND SAMPLER II VON G-DATA

Die Arbeitsweise des GDATA-Samplers ist etwas anders als die der ersten zwei Geräte. Von vornherein gibt es zwei verschiedene Wandler und auch zwei verschiedene Software Versionen, die man getrennt erwerben muß. Zwar hat man dadurch die freie Wahl der Geräte, muß allerdings etwas tiefer in die Tasche greifen.

Die Hard- und Software ist untereinander kompatibel und bietet, wie vom Preis her zu erwarten, auch unterschiedliche Leistungen. Wir haben uns für die bessere Version entschieden, da sie sich, hauptsächlich aufgrund der Software, positiv von der Konkurrenz unterscheidet.

Der A/D-D/A Wandler besteht aus einem 12X15 cm grossen Kästchen, das mit einem ca. 80cm langen Kabel an den Druckerport angeschlossen wird. Es ist daher flexibel und kann auch abseits des Rechners stecken. Der Druckerport wurde durchgeschliffen; wodurch das Umstecken des Druckerkabels entfällt. An der Rückseite des Samplers stehen drei 3,5 mm-Buchsen (Eingang, Ausgang und Netz) zur Verfügung. Die Versorgungsspannung liefert das mitgelieferte Netzteil.

Auf der Frontseite befinden sich drei Regler: Lautstärke Eingang, Lautstärke Ausgang sowie ein Höhenfilter, der störende Frequenzen aussiebt.

Weiterhin ist ein Peek-Indikator vorhanden, mit dem man ohne viel Probieren die Aussteuerung einstellen kann, ein Resettaster, der den momentanen Programmablauf unterbricht und ein Schalter zum Umschalten zwischen Aufnahme und Wiedergabe. Alles in allem eine anspruchsvolle Hardware, die zusätzlich noch sehr ansprechend gestaltet ist. Doch auch die einfache Hardware-Version ist nicht zu unterschätzen. Sie unterscheidet sich von der obigen Version lediglich durch den fehlenden Peek-Indikator, den Ausgangsregler und den Höhenfilter. Bastler können sich dieses Gerät auch selbst bauen, denn in der einfacheren Softwareversion ist der komplette Schaltplan dafür enthalten.

TEAC-Diskettenlaufwerke für Atari ST:

vollkompatibel, leise, 12 Monate Garantie, anschlußfertig, inkl. Kabel, Netzteil, Metallgehäuse. TEAC FD 35 FN bzw. FD 55 FR. Siehe auch Test in 68000er 8/87.

3,5" 720 KB **298.-** — Doppelstation 1,4 MB: **589.-**

5,25" 720 KB **369.-** — 40/80 Tr. Umschaltung optional

Ab sofort lieferbar: Doppelstation 1,4 MB, bestückt mit 2 x 1037 A, Superpreis **549.-**

Brandneu: NEC 1037

Superflaches Diskettenlaufwerk mit einer Bauhöhe von nur 25,4 mm und einer Kapazität von 1 MB (unformatiert). Anschlußfertig und vollkompatibel. **298.-**

Alle Laufwerke auch als Bausatz (Netzteil vormontiert) lieferbar. Preisnachlaß 30.–

fse Frank Strauß Elektronik, St. Marienplatz 7, Tel. 0631 / 16258
6750 Kaiserslautern

MUC Calc

Firmencontrolling

MUC Calc ist ein Softwareprodukt zur Überwachung und Planung Ihrer Firmenfinanzen über einen Zeitraum von 5 Jahren. MUC Calc wurde speziell aufgrund der Anforderungen von kleineren und mittelständischen Unternehmen entwickelt und getestet.

- Monats-, Quartals- und Jahresüberwachung/Planung der Firmenfinanzen über 5 Jahre
- Dateneingabe und Datenübersicht in Tabellenform und
- **automatische grafische** Aufbereitung aller relevanten Firmendaten (Kuchen-, Balken- und Liniengrafik) auf Bildschirm und Drucker
- **automatische Datenübernahme** aus Programmen der fibuMAN Reihe (fibuMAN e,f,m)
- auf Wunsch spezielle Anpassung an Ihre Firma
- MUC Calc läuft mit SW Monitor auf allen Atari ST Konfigurationen (auch MEGA ST und Harddisk)
- MUC Calc läuft vollständig unter GEM und wurde besonders hinsichtlich einfacher Bedienung ausgelegt.
- kein Kopierschutz, Updateservice

Fordern Sie unverbindlich das **MUC Calc Info** an.

DEMO Disk mit Handbuch **60 DM** (wird beim Kauf angerechnet – nur gegen V-Scheck oder NN)

MUC Calc kompl. mit deutschem Handbuch/Diskette **398 DM**

1st PROPORTIONAL

(siehe Testbericht ST-Computer 6/87)

1st PROPORTIONAL ermöglicht Ausdrucke von 1st Word/1st Word Plus Texten in **Proportionalschrift** im **Blocksatz**. Dadurch erhält man Ausdrucke, die gesetzten Texten in nichts nachstehen. **1st PROPORTIONAL** unterstützt alle proportionalschriftfähigen Nadeldrucker (9, 18, 24 Nadeln) und proportionalschriftfähige Typenraddrucker (mit entsprechendem PS-Typenrad).

- läuft vollständig unter GEM (SW-Monitor), unterstützt alle Optionen von 1st Word Plus
- verschiedene Zeilenlineale in **verschieden breiten Proportionalschriften im Blocksatz!**
- im Text integrierte Grafik kann in 3 verschiedenen Größen gedruckt werden, dadurch Ausdruck exakt wie auf dem Bildschirm oder **unverzerrter** Ausdruck (Kreise bleiben rund, auch bei 24 ND!)
- Grafiktreiber für 9, 18, 24 Nadeldrucker, dadurch endlich satte Schwärzen auch auf 24 Nadeldruckern
- Verkettung von bis zu 25 Texten im Ausdruck bei wahlfreien Druckoptionen für jeden Text und automatischer Durchnummerierung; wichtig bei umfangreichen (wissenschaftlichen) Arbeiten.
- **1st PROPORTIONAL** wird mit Druckertreibern für NEC P5/6/7, TA GABI 9009, Brother HR 15, HR 25, 35, EM 701, 711, 811, LQ 800, Seikosha SL 80 AI, STAR NL 10, FX 85, auf einer einseitigen Diskette mit 60-seitigem, deutschen Handbuch geliefert (weitere Druckeranpassungen auf Anfrage).
- 7-seitiges Info mit Probeausdrucken anfordern – (bitte 2,00 DM in Briefmarken beilegen)

1st PROPORTIONAL kostet nur **95 DM**

Hans Christian Knib · Viktoriastr. 9 · 5100 Aachen · 02 41 / 5 47 34

Modula II-Dev. V3.0 . 294,95 DM

Cambridge Lisp	DM 299,–
Lattice C V3.04	DM 194,95
MCC-Make	DM 99,95
Personal Money M.	DM 59,95
Deep Space	DM 59,95
Electronic Pool	DM 42,95
MCC-Pascal V2.0	DM 175,95
MCC-Bcpl-Compiler	DM 189,–
Cornerman	DM 59,95
Q-Ball	DM 52,95
Terristal Encount.	DM 32,95
Psion Chess	DM 59,95
Central Point – Copy II/ST-Kopierprogramm	DM 99,–
MCC-Assembler	**DM 99,95**

Kostenlose Prospekte von...



Computerversand C W T G
Joachim Tiede
Bergstraße 13 · 7109 Roigheim
0 62 98 / 30 98 von 17-19 Uhr

HÄNDLERANFRAGEN ERWÜNSCHT!

GE-Soft

COMPUTER

520 STM incl. Maus u. ROM	568,-
Monitor SM 124	448,-
520 STM incl. Maus u. Rom, Monitor SM 124	998,-
1040 STF incl. Maus, ROM, Monitor	1498,-

LAUFWERKE

JVC Diskettenlaufwerk 2 x 40 Track im Stahlblechgehäuse 5 1/4" incl. Netzteil anschlußfertig für ATARI	nur 298,-
NEC 1036 A anschlußfertig für Atari im Stahlblechgehäuse	328,-
NEC 1036 A wie oben, jedoch als Bausatz mit sämtl. Kabeln, etc.	nur 298,-
3 1/2" Laufwerk NEC 1036A wie oben; jedoch als A- oder B-Lauferk (schleifbar)	nur 398,-

FESTPLATTE

20 MB – bootfähig Gehäuse gleichzeitig als Monitorständer nutzbar, erweiterbar auf 40 MB, lieferbar Mitte November	1.098,-

DRUCKER

STAR NL 10	nur 548,-
TA TRD-170 S-Typenraddrucker incl. Treiber-Software für ST - anschlußfertig	nur 888,-

KABEL

Atari ST auf Scart	nur 49,80
Atari auf Chinch	nur 44,80

SPEICHERERWEITERUNGEN

Aufrüstsatz auf 1 MB	nur 128,-
Platine – teilsteckbar auf 1 MB	nur 198,-

DISKETTEN

Fuji 10 St.	nur 29,95

GE-Soft 5300 Bonn 1, Graurheindorferstr. 9, 02 28 - 69 42 21

Public-Domain Softwarepakete

★ 5 Markendisketten MF 1 DD, doppelseitig formatiert und gefüllt mit guter Public-Domain Software



Paketpreis nur DM 45,–

★ *Weihnachtsangebot* ★
3 Pakete nur DM 119,–
6 Pakete nur DM 219,–

Paket	Inhalt
1	PD01 – PD10 aus ST-Computer
2	PD11 – PD20 aus ST-Computer
3	PD21 – PD30 aus ST-Computer
4	10 PD-Disk aus Kalifornien
5	10 PD-Disk aus Kalifornien
6	10 PD-Disk aus aller Welt
7	10 PD-Disk aus Kalifornien
8	10 PD-Disk aus Kalifornien
9	10 PD-Disk aus Kalifornien
10	PD 31-PD 40 aus ST-Computer
11	PD41-PD51 aus ST-Computer
12	10 PD-Disk aus aller Welt
13	10 PD-Disk aus Kalifornien
14	10 PD-Disk aus Kalifornien
15	10 PD-Disk aus Kalifornien

Paket 16: Enthält PD 52 – 61 aus ST-Computer. Fugger ★ Ausland ★ Kepler ★ Eti-Master ★ Trio ★ Kerne ★ Meßwert, etc...

Paket 17: Enthält PD 62 – 71 aus ST-Computer. E-Plan ★ Paintlux ★ Degas Elite ★ Fontmaker ★ Vang Gogh ★ Orbit ★ Roulette u.v.m.

Paket 18: ★ Label, schönes Etikettprg. ★ GfA + Logo Prgs ★ Meteosat Diashow ★ Kissdemo ★ Heinzelmann ★ Trek 2 Textadventure ★ Miami Digisound (1MB) ★

Paket 19: Enthält PD 72 – 81 aus ST-Computer. Z. B. Videoarchiv ★ Maxidisk ★ Datebook ★ Periode ★ AESLIB ★ U_Boot ★ Fonts für PD 40 ★ Malprogramm ★ etc...

Paket 20: Enthält PD 82 – 91 + Updates „U1" und „U2" aus ST-Computer, z. B. Skat, Grusel, Masterpainter, Starofix, Hypervoc, etc...

Paket 21: Enthält PD 92 – 101 aus ST-Computer. Z. B. ST-Speech ★ ST-Klick ★ MACPAN ★ Andurlin ★ Minigolf ★ Trash ★ Sharp ★ Minenfeld ★ Missile und vieles mehr

Paket 22: Enthält PD 102-111 aus ST-Computer. Z.B. Lohnsteuer * Aktie * Depot * Poster * Napoleon * Druckerutilities * Wordplus Utilities * Soundcomputer...

Paket 23: Enthält 12 Disks aus Kalifornien. Protect schützt vor Viren * NeoCalender erstellt Kalender aus Ihren Bildern * neue Borders für Printmaster * ArtDeco Fonts für Degas Elite * Bannermaker * Turtel Harddiskbackup * Spiele etc....

Gewünschte Pakete ankreuzen u. Bestellung einsenden an:

IDL-Software
– Public Domain –
Alkmaarstraße 3 · 6100 Darmstadt 13

- [ ] Info-Diskette Nr. 4 DM 5,- PD-Liste + Clip Art für Publishing Partner
- [ ] NACHNAHME (Nur Inland, zuzügl. DM 5,– NN-Gebühr)
- [ ] Scheck über DM __________ liegt bei,

Preise wie noch nie !!!!

Durch tägliche Aktualisierung der Preise !!!!

GFA-Basic/Compiler DM 87,00.
Alle Games mit mind. 20% Rabatt
Paderc.FL1 279.-512K Weide **225**.-
CompTec GbR, Grüner Weg 2a
3584 Zwesten, 05626/569+1431
HARD+SOFTWARE ATARI ST

Die Software

Spätestens nach dem Laden wird klar, daß das Programm so einiges zu bieten hat: im Vergleich zu den anderen Systemen nämlich wesentlich mehr Komfort und Feinheiten. Sogar der Anschluß eines MIDI-Keybords wurde eingebaut. Aber alles der Reihe nach.

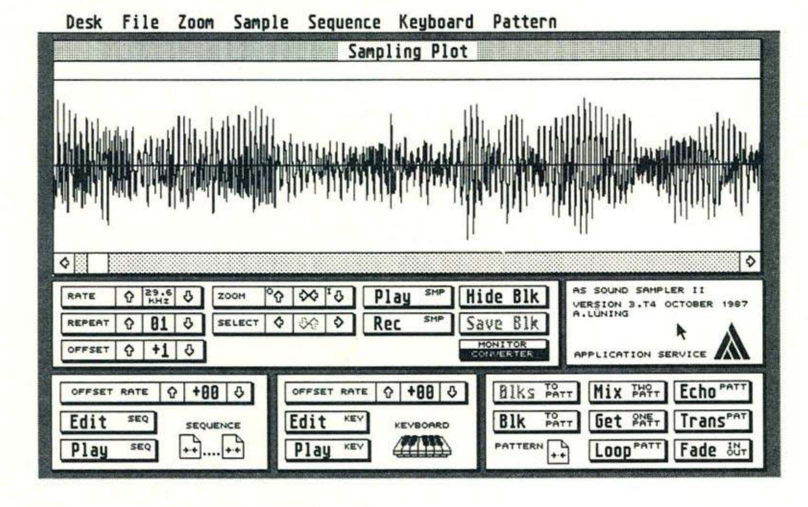


Bild 3: Der AS SOUND SAMPLER II

einzelnen Pattern kann man, in einer Ablaufliste zu einer Sequenz vereinen. Die angegebene Pattern werden der Reihe nach abgespielt, und zwar jedes für sich mit variabler Geschwindigkeit und unterschiedlicher Anzahl. Somit kann man durch Wiederholen einzelner Pattern ein minutenlanges Musikstück generieren,

Zur Aufnahme

Hier gibt es nichts Neues, bis auf die Tatsache, daß der im Gerät vorhandene Lautstärkeregler in Kombination mit dem PeekIndikator sich zur Aussteuerung wesentlich besser eignet als ein Lautstärkeregler an der Tonquelle. Es ist eben doch einfacher, dies gleich neben dem Rechner einzustellen, als an einer acht Meter entfernten Stereoanlage.

Die Verarbeitung

Hier liegt zweifelsohne die Stärke dieses Programms. Aus einem Sample kann man einen Bereich sehr genau ausschneiden (die Lupe machts möglich), und diesen als Pattern ablegen; der Rest dieses Samples kommt gnadenlos in den Mülleimer. Ein Patter kann nun vielfältigen Bearbeitungen unterzogen werden: Mischen mit anderen Pattern, Unterlegen mit Echo und Aus- bzw. Einblenden ist kein Problem.

Jetzt ist aber noch nicht Schluß, das Beste kommt noch: Die

oder sich eine Politikerrede aus einzelnen Worten zusammenbasteln. Wie schon erwähnt, soll es ja gewisse Musiker geben, die nach diesem Prinzip ihre Platten produzieren.

MIDI spielt mit

Mit einem MIDI-fähigen Keybord lassen sich der Software noch weitere Features entnehmen. Man kann nämlich einzelne Pattern auf bestimmte Tasten legen und diese per Tastendruck abspielen.

Weiterhin läßt sich ein bestimmter Ton, welcher auch irgendein kurzes Geräusch sein kann, auf alle Notentöne umrechnen und auf das Keybord legen. Als Beispiel hierfür könnte man sich den Sound einer E-Gitarre vorstellen, der dann, nachdem er in der Tonhöhe umgerechnet wurde, per Klaviertastatur aus dem Rechner ertönt. Oder auch mit dem Bellen eines Hundes – hier ist nur Ideenreichtum gefragt. Leider bleiben einem diese Features ohne MIDI-Keybord verschlossen, doch bietet der GDATA-Sampler auch ohne dies die besten Möglichkeiten zur Soundverarbeitung.

AC Sound Sampler II ist ein hervorragendes Paket, das sowohl aufgrund des Hard- als auch des Software-Konzepts im privaten sowie im semiprofessionellen Bereich seinen Einsatz finden kann. Ein Manko jedoch ist, daß kein Handbuch beiliegt sondern nur ein File auf Diskette. Es wird jedoch auf ein käufliches Buch verwiesen, das diese Lücke füllen soll und laut GDATA günstig über sie zu erhalten ist.

FAZIT

Dieser Test zeigt deutlich die Unterschiede der Systeme, vor allem in der Software, denn die Samplequalität der Hardware ist aufgrund der 8 Bit-Struktur bei allen ähnlich, nur eben der Bedienungskomfort ist unterschiedlich.

Jetzt kann nur der Zweck Ihrer Verwendung darüber entscheiden welches Gerät sie bevorzugen. Das Einbinden von Tönen in kleine Programme verlangt nicht mehr, als das was PRO SOUND und ST REPLAY bieten. Der Preis spricht für PRO SOUND. Wer aber nicht nur kurz samplen will, sondern damit experimentieren und wie oben beschrieben mehrere Samples zu einer neuen Sequenz zusammenstellen will, oder den Rechner sogar auf der Bühne zum Einsatz bringen sollte, der muß, obwohl dies die teuerste Lösung ist, zwangsweise mit dem Sampler und vor allem mit der Software von GDATA liebäugeln.

Philippe Lerailler

EIN SOUND-SAMPLER ? JA, ABER WOZU?

Die Anwendergruppe läßt sich nicht klar bestimmen. Es gibt deren verschiedene, zum Beispiel den Programmierer, der einen Sampler dazu benutzt, seine eigenen Programme zu verzieren. Man stelle sich nur vor, daß der Computer sich mit der Stimme seines Besitzers zu Worte meldet und Fehlermeldungen plötzlich in gesprochenem Klartext ausgibt.

Ich kenne mehr als einen, der darüber stolz wäre. Oder es wäre auch denkbar, einen ST zur sprechenden Uhr umzufunktionieren. Musik, Geräuche, gesprochene Meldungen sind eine Reihe von Sachen, die Ihre Programme attraktiver machen.

Andere Benutzer sind die Soundbastler, die keine Ruhe haben, solange sie nicht den letzen Hit von Michael Jackson mit der 5. von Ludwig van Beethoven gekreuzt oder unglaubliche Wörter in der Mund von Helmut Kohl geschoben haben.

Die erzielbaren Effekte sind groß. Man denke nur daran, wie leicht sich mit drei hintereinandergequirlten Soundsequenzen ein neues Lied von Modern Talking gestalten läßt, womit ich die Leistungen dieser Gruppe jedoch keineswegs schmälern möchte (eventuell sollte man es einmal rückwärts abspielen, vielleicht ...).

Jedenfalls ist ein Sound-Sampler dafür eine sehr große Hilfe, denn er erlaubt es, Töne viel besser zu verarbeiten als mit einem Tonband.

Für die Extremform der Anwendung in dieser Richtung steht der Technomusiker, der die Musik nur noch als eine Folge von Bytes versteht. Der Soundsampler wird ihm helfen, elektronische Musik zu produzieren, die "wie die echte klingt".

Vielleicht kennen Sie noch andere Anwendungen für die SoundSampler. Die Fortschritte der Technik in diesem Bereich haben dazu beigetragen, daß sie heute jedem Computerbesizer zugänglich ist.

Viel Spaß beim Samplen !!!

EIN WORT ÜBER "SAMPLING"

Ein Deutsch-Englisch Wörterbuch spricht zum Thema 'Sampling' von Proben und Mustern!!

Das klingt keineswegs technisch. Dabei ist diese Übersetzung gar nicht so falsch, wie man es vermutet. Proben sind schließlich kleine Mengen von in großen Mengen produzierten Waren. Beim Sound-Sampling wird der Sound (als große Menge) in ganz vielen winzigen Teilen verarbeitet (den Proben!!).

So kommt man hinter die Technik des Sound-Samplings: Töne werden in erster Linie durch Luftdruckschwankungen und ferner durch elektrische Spannungsschwankungen (zum Beispiel in Ihrer HiFi-Anlage) produziert. Diese Töne sind für das menschliche Ohr hörbar, doch kann kein Computer mit diesen kontinuierlichen (analogen) Größen etwas anfangen. Jetzt erscheint der Sampler, dessen Aufgabe es ist, Analoges in Digitales zu wandeln (und umgekehrt). Stellen Sie sich die Sportaufnahmen vor, deren Ablauf in einzelne Abschnitte (Zeitlupe) zerlegt worden sind. Ebenso werden die Spannungen mehr als tausend Mal pro Sekunde gemessen und die so gewonnenen Meßwerte gespeichert.

Eine wichtige Komponente dabei ist die Samplingrate, also die Anzahl der Messungen in einer bestimmten Zeit. Je öfter gemessen wird, desto besser, feiner wird der Ablauf, der Klang.

Dabei tritt die Tatsache in Erscheinung, daß eine bestimmte Frequenzbreite (z.B. bis 20 KHz für das menschliche Ohr) die doppelte Abtastrate (also 40 KHz) braucht, um Töne in ihrer Charakteristik wiederzugeben. Ein guter Beweis hierfür ist die CD-Technik, sie arbeitet mit einer Samplingrate von 44,1 KHz.

Je öfter natürlich gesamplet wird, desto zahlreicher sind die Informationen, und desto mehr Speicherplatz wird benötigt. Das Resultat: Je besser die Qualität, desto kürzer die Zeit, die man aufnehmen kann. Nehmen wir beispielsweise eine Samplingrate von 20 KHz, so benötigt eine Sekunde Aufnahmezeit bei einer Auflösung von 8 Bit (1 Byte) exakt 20 KByte Speicherplatz. Geht man nun davon aus, daß man etwa 800 KByte freien Speicher zur Verfügung hat, so kann man 40 Sekunden (800/20=40) lang samplen. Die damit aufzeichenbare Frequenz liegt bei maximal 10 KHz.

Bei einer geringeren Samplerate, z.B. 10 KByte, wäre die Zeit logischerweise doppelt so lang (800/10=80), wobei allerdings nur noch Frequenzen unter 5 KHz wiedergegeben werden können.

Man muß hierbei zwangsweise einen Kompromiß zwischen Aufnahmedauer und Aufnahmequalität eingehen?

Die technische Seite

Wie die Information nun in den Rechner kommt, ist eine andere Angelegenheit, der wir uns auch kurz widmen wollen. Da es sich bei

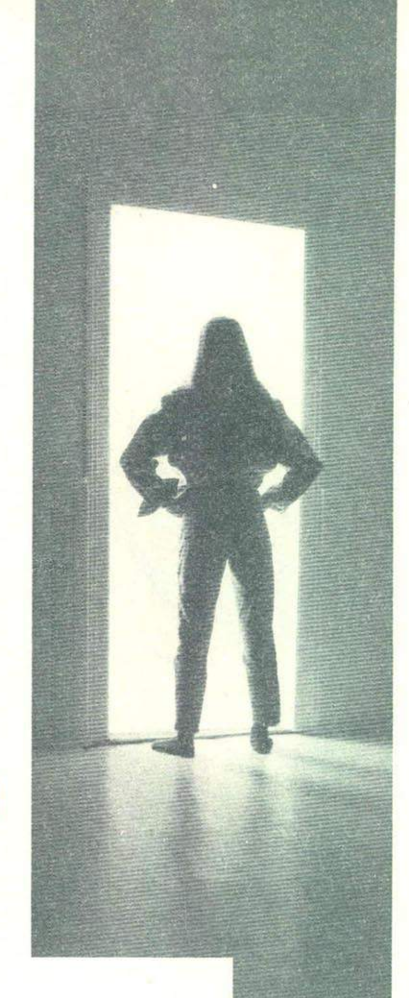
Sprache oder Musik, wie schon oben erwähnt, um ein analoges Signal handelt, muß dieses zuerst in ein digitales, also dem Rechner verständliches Signal umgeformt werden. Dazu benötigt man einen A/D-Wandler (Analog nach Digital). Das daraus gewonnene digitale Signal versteht nun der Rechner und kann damit arbeiten. Bei der Aufnahme macht er dann nichts anderes, als die Informationen hintereinander in seinem Speicher abzulegen, und zwar solange es ihm aufgetragen wurde oder bis er voll ist.

Beim Abspielen werden nun diese Informationen wieder ausgegeben. Dies kann im einfachsten Fall über den eingebauten Monitorlautsprecher erfolgen. Dabei tritt aber ein starker Qualitätsverlust auf, denn die Ausgabe erfolgt über den Soundchip des ST und zwar nur über Änderung der Lautstärke eines Dauertones. Da der Soundchip dies bekanntlich nur in 16 Stufen kann, muß die 8 Bit-Information auf 4 Bit gekappt werden ($2^4 = 16$). Wenn man Musik, Sprache oder Geräusche in ein eigenes Programm einbauen möchte, ist dies aber durchaus ausreichend.

Die zweite Möglichkeit ist, die im Rechner gespeicherten Informationen wiederum über einen Wandlerbaustein umzusetzen. Dazu gibt es den D/A Wandler (von Digital nach Analog). Man erhält dadurch wieder ein analoges Signal, was man nun über die Stereoanlage anhören kann. Die damit erreichte Klangqualität ist, wenn man bedenkt, daß bei einem 8 Bit Sampler die Musik nur in lediglich 256 verschiedene Stufen unterteilt werden kann ($2^8 = 256$), schon erstaunlich gut. Ein CD-Player schafft es da immerhin schon auf 65536 verschiedene Stufen ($2^{16} = 65536$). Daß solch ein fein abgetastetes Signal wesentlich besser klingt, ist wohl jedem klar. Natürlich soll der Rechner auch keinen CD-Player immitieren, obwohl es auch 16 Bit Sampler für den ST gibt. Diese liegen jedoch in einer anderen Preisklasse als unsere drei Geräte und fallen somit nicht in diese Gruppe.

	PRO SOUND DESIGNER	**AS SOUND**	**ST REPLAY**
Hersteller	Eidersoft	G Data	Microdeal
Rechneranschluß	Druckerport	Druckerport	Romport
Stromversorgung	9V-Batterie	Mitgeliefertes Netzteil	Über Rechner
HiFi Anschluß	3,5mm Klinke*	3,5mm Klinke	Cinchbuchse
Regelbarer Ein/Ausgang	nein/nein	ja/ja	nein/nein
Externer Peekindicator	nein	ja	nein
Auflösung	8 Bit	8 Bit	8 Bit
Aufnahmefrequenz	30-3 KHz	36.4-11.3 KHz	31-5 KHz
Aufnahme dauer bei 1 MByte RAM	25-256 Sek	24-77 Sek	30-190 Sek
bei 20Khz und 1 MByte RAM	38 Sek	43 Sek	46 Sek
Mitgelieferte Programmbeispiele	ST-Basic Fast Basic C	GFA Basic OMIKRON - Basic C	ST-Basic Fast Basic C Modula
Preis	DM 165.- (*)	Hardw. DM 249.- Softw. DM 149.-(**)	298.-
Vertrieb	CCD	G Data	Print Technik Knupe
Remarks	* Mit HiFi Ausgang als Option ** DM 169.- für fremde Systeme		

Auf der Schwelle zum Licht



Das Geheimnis des GEMDOS Teil II

Die Fehlermeldungen des TOS (Teil 1)

In dieser Folge geht es um die Fehlermeldungen, die die TOS-Systemfunktionen liefern, sowie um die Möglichkeiten des TOS, diese anzuzeigen. Dabei machen wir sogar Abstecher in das BIOS und das AES.

Ich hoffe, daß Sie nach Lektüre dieses Artikels ein wenig mehr mit den Fehlercodes anfangen können, die Ihnen GEMDOS beim Programmieren bisher immer fleißig geliefert hat.

BIOS-Fehlermeldungen

Viele BIOS-Funktionen geben Fehlermeldungen zurück, die Aufschluß über die Art des aufgetretenen Fehlers geben sollen. In den Atari-Dokumentationen haben sie symbolische Namen bekommen, die man sich besser merken kann als die nackten Zahlenwerte, weswegen ich diese Bezeichnungen auch in dieser Serie benutzen werde (Tab.1). Man kann sich diese Konstanten z.B. in eine Include-Datei schreiben und zur Fehlerbehandlung in eigenen Programmen verwenden. Atari hat die Werte -1 bis -31 für BIOS-Fehlercodes reserviert.

Da es sich dabei hauptsächlich um Fehler beim Disketten-Zugriff handelt, machen wir einen kleinen Ausflug in den Disk-Treiber des BIOS. Das Wissen um die Bedeutung und Anwendung der erwähnten BIOS-Funktionen wird vorausgesetzt und findet sich in diversen Büchern. Die Disketten-Funktionen des BIOS gliedern sich in zwei Ebenen. Zur unteren Ebene gehören die Spur-/Sektor-orientierten Funktionen 'Floprd', 'Flopwr', 'Flopfmt' und 'Flopver' des XBIOS. Von diesen Funktionen ist bei der Erklärung der Fehlermeldungen die Rede, wenn von Lesen, Schreiben, Formatieren und Verifizieren gesprochen wird.

Auf der oberen Ebene befinden sich die BIOS-Funktionen 'Rwabs', 'Mediach' und 'Getbpb', die nicht nur für Diskettenlaufwerke vorgesehen sind, sondern auch für Harddisks und RAM-Disks zuständig sind. Bei 'Getbpb' ist zu beachten, daß alle auftretenden Fehler nur durch 0L gemeldet werden, also keine weitere Differenzierung der Fehlerursache mehr erfolgt.

Fehler können nun auf beiden Ebenen auftreten, wobei i.a. jede Ebene ihre eigenen Fehlermeldungen hat. Einige Fehlercodes der unteren Ebene resultieren direkt aus den Fehlermeldungen des FDC (Floppy Disk Controller), deren Erläuterung sich im "Floppy-Kurs" (ST-Computer 11/87) findet.

In einigen Fällen wurde die eindeutige Benutzung der Fehlercodes nicht ganz durchgehalten, d.h., gleiche Fehler erzeugen unterschiedliche Meldungen.

ERROR

Zum einen zeigt diese Meldung einen DMA-Fehler beim Lesen oder Formatieren (nicht beim Verifizieren) an. Achtung: Beim Schreiben wird ein DMA-Fehler überhaupt nicht erkannt!

Außerdem gibt er einen Timeout beim Schreiben oder Formatieren (nicht beim Lesen oder Verifizieren) an, d.h.

wenn ein Kommando des Floppy Disk Controllers (FDC) nicht innerhalb einer gewissen Zeitspanne ausgeführt werden konnte.

EDRVNR

Hiermit wird ein Timeout beim Lesen angezeigt. Bei einem 'Rwabs' ohne angeschlossene Floppy tritt dieser Fehler ebenfalls auf. Aber ganz ohne Floppy arbeitet es sich ja sowieso recht schlecht.

E–CRC

Hierdurch wird auf einen Prüfsummen-Fehler aufmerksam gemacht, was auf einen Lesefehler hindeutet. Die Meldung korrespondiert mit dem CRC-Bit des FDC-Statusregisters.

Dies gilt jedoch nur für das Blitter-TOS. Beim alten TOS wird der CRC-Fehler zwar korrekt erkannt, aber stattdessen eine andere Fehlermeldung zurückgegeben (EREADF bzw. EWRITF bei Lese/Verifizier- bzw. Schreib-/Formatier-Operationen).

E–SEEK

Wenn das BIOS eine Spur auch nach mehreren Versuchen nicht ansteuern kann, bricht es die Disketten-Operation hiermit ab. Vermutlich ist die Diskette auf dieser Spur nicht formatiert oder defekt.

ESECNF

Wenn der FDC einen gesuchten Sektor auf der Diskette nicht finden kann, wird diese Meldung erzeugt. Der Fehler entspricht dem Record-Not-Found-Bit des FDC-Statusregisters. Entweder ist die Diskette nicht ausreichend formatiert, oder sie ist beschädigt.

EPAPER

Das soll anzeigen, daß der Drucker nicht betriebsbereit ist, weil kein Papier eingelegt ist. Da der Atari hardwaremäßig das entsprechende Signal auf dem Centronics-Port gar nicht abfragen kann, ist diese Fehlermeldung wohl – wenn überhaupt – eher zukünftigen STs vorbehalten.

EWRITF

Dieser "allgemeine Schreibfehler" tritt beim Schreiben oder Formatieren auf, wenn der FDC einen "Lost Data-Fehler" meldet. Auch dies funktioniert so nur beim Blitter-TOS; beim alten TOS wird stattdessen ein E–CRC gemeldet.

EREADF

Der "allgemeine Lesefehler" entspricht EWRITF beim Lesen oder Verifizieren (auch Vertauschung mit E–CRC beim alten TOS).

Beim alten TOS wird ein "Lost Data" beim Lesen jedoch (nicht beim Verifizieren) ignoriert, was einem Kopierschutz, der darauf baut, beim neuen TOS Schwierigkeiten macht.

Desweiteren bekommt man diesen Fehler auch bei einem DMA-Fehler oder Timeout beim Verifizieren.

WPRPRO

Hiermit wird auf einen versuchten Schreibzugriff bei aktiviertem Schreibschutz aufmerksam gemacht.

E–CHNG

'Rwabs' zeigt mit dieser Meldung an, das ein definitiver Wechsel des Speichermediums (z.B. Disketten) stattgefunden hat.

Die untere Ebene der Disketten-Funktionen kümmert sich nicht um Disketten-Wechsel, da die Routinen zur Diskettenwechsel-Erkennung und -Behandlung zur oberen Ebene gehören.

EUNDEV

Das angesprochene Laufwerk ist unbekannt. Dieser Fehler tritt nur bei 'Rwabs' und 'Mediach' auf, bei 'Getbpb' wird hierbei 0L zurückgegeben. Ein Laufwerk ist unbekannt, wenn sich kein Treiber dafür in diese Disk-Routinen eingehängt hat. Die Floppy-Laufwerke A: und B: sind immer bekannt (s. auch EDRVNR). Der Zustand der Systemvariablen 'drvbits' hat hiermit nichts zu tun.

EBADSF

Die bisher beschriebenen Fehlermeldungen beim Verifizieren erhält man nur, wenn man das Verifizieren direkt über die XBIOS-Funktion aufruft. Die Verifizier-Routine wird jedoch auch von 'Rwabs' und der Formatier-Funktion 'Flopfmt' benutzt. Dabei wird bei einem Fehler jedoch einfach ein pauschales EBADSF zurückgegeben.

EOTHER

Hierbei handelt es sich nicht um eine Fehlermeldung, sondern um eine Anfrage. Das BIOS arbeitet bei nur einem Disketten-Laufwerk ja bekanntlich mit zwei logischen Laufwerken A: und B:. Wird das jeweils andere logische Laufwerk angesprochen, fordert das BIOS hiermit zum Diskettenwechsel auf. Normalerweise wird dies vom Critical Error Handler behandelt, so daß diese Fehlermeldung nicht an den Aufrufer der BIOS-Funktion zurückgegeben wird.

In Tab.1 findet der aufmerksame Leser noch die Fehlercodes EUNCMD, EBADRQ und EMEDIA; außerdem fehlt die -12. Diese Meldungen werden – soweit mir bekannt – vom BIOS aber garnicht verwendet.

Name	Wert	Kurz-Bedeutung
E_OK	0	OK (kein Fehler)
ERROR	-1	Allgemeiner Fehler (kann alles heißen)
EDRVNR	-2	Laufwerk nicht bereit
EUNCMD	-3	Unbekanntes Kommando
E_CRC	-4	Prüfsummen-Fehler
EBADRQ	-5	Kommando in der Form nicht möglich
E_SEEK	-6	Spur nicht gefunden
EMEDIA	-7	Unbekanntes Medium
ESECNF	-8	Sektor nicht gefunden
EPAPER	-9	Kein Papier mehr
EWRITF	-10	Schreibfehler
EREADF	-11	Lesefehler
EWRPRO	-13	Schreibversuch trotz Schreibschutz
E_CHNG	-14	Medium-Wechsel aufgetreten
EUNDEV	-15	Unbekanntes Gerät
EBADSF	-16	Fehler beim Verifizieren
EOTHER	-17	Andere Diskette einlegen (Anfrage)

Tab. 1 – BIOS-Fehlercodes

GEMDOS-Fehlermeldungen

Die meisten GEMDOS-Funktionen geben einen Wert zurück, der anzeigt, ob sie ordnungsgemäß ausgeführt werden konnten, oder ob ein Fehler auftrat, ähnlich wie beim BIOS. Für die Fehlermeldungen des GEMDOS sind dabei 32-Bit-Integers im Bereich von -32 bis -127 vorgesehen (einige Funktionen liefern nur 16-Bit-Werte zurück!). Funktionen, die Datei-Zugriffe machen, können außerdem die nun schon bekannten BIOS-Fehlermeldungen zurückgeben. Eine Auflistung mit den symbolischen Bezeichnungen und einer "Kurzbeschreibung" findet sich in Tab.2. In einigen Dokumentationen ist ENMFIL fälschlicherweise als -47 angegeben.

Name	Wert	Kurz-Bedeutung
E_OK	0	OK (kein Fehler)
ERROR	-1	Allgemeiner Fehler
EINVFN	-32	Ungültige Funktionsnummer
EFILNF	-33	Datei nicht gefunden
EPTHNF	-34	Pfad nicht gefunden
ENHNDL	-35	Keine Handles mehr frei
EACCDN	-36	Zugriff nicht möglich
EIHNDL	-37	Ungültiges Handle
ENSMEM	-39	Zu wenig Speicherplatz
EIMBA	-40	Ungültige Speicherbereichs-Adresse
EDRIVE	-46	Ungültige Laufwerkskennung
ENSAME	-48	Nicht die gleichen Laufwerke
ENMFIL	-49	Keine Dateien mehr gefunden
ERANGE	-64	Datei-Zeiger aus gültigem Bereich
EINTRN	-65	Interner Fehler
EPLFMT	-66	Ungültiges Programm-Format
EGSBF	-67	Speicherbereich nicht vergrößerbar

Tab. 2 – GEMDOS -Fehlercodes

Eigentlich sollte jeder Fehlermeldung eine bestimmte, eindeutige Bedeutung zukommen. Was für das BIOS noch einigermaßen zutrifft, gilt für GEMDOS jedoch noch lange nicht. Man darf auf Grund dieser Tabelle nämlich nicht davon ausgehen, daß nun das Thema "GEMDOS-Fehler" schon erledigt ist. Die Programmierer des GEMDOS haben die Fehlercodes nämlich manchmal recht wahllos verwendet.

Oft wird eine Fehlerursache, die bei mehreren Funktionen auftreten kann, dem aufrufenden Programm durch unterschiedliche Fehlercodes mitgeteilt, in einigen Fällen kann sogar ein Fehler bei ein und derselben Funktion zu verschiedenen Fehlermeldungen führen.

Ein gutes Beispiel hierfür ist der Mangel an freiem "internen Speicher" (s. Speicherverwaltung), der bei vielen Funktionen auftreten kann, aber nur in den seltensten Fällen zur Fehlermeldung ENSMEM führt, sondern meistens in einem ERROR, EPTHNF, EFILNF, ENMFIL, EDRIVE oder sogar in Bömbchen resultiert.

Andererseits kann ein Fehlercode die unterschiedlichsten Ursachen haben. Ein gutes Beispiel hierfür ist EACCDN, das anscheinend für fast alles herhalten muß, was nicht durch andere Fehlermeldungen abgedeckt ist.
Dies erschwert natürlich die einheitliche Behandlung von GEMDOS-Fehlern durch eine globale Fehlerbehandlungs-Routine, da die Bedeutung der Fehler von der aufgerufenen Funktion abhängt. Dies steht im Gegensatz zur offiziellen GEMDOS-Dokumentation, wo der Eindruck erweckt wird, als ob die meisten allgemeinen Fehler gleich behandelt werden könnten.

Das soeben beschriebene Fehler-Chaos kommt übrigens dadurch zustande, daß die von den GEMDOS-Routinen der unteren Ebenen gelieferten Fehlercodes von den übergeordneten Funktionen "verfälscht" weitergegeben werden.

Desweiteren werden einige Fehler überhaupt nicht bemerkt oder nur sehr unvollständig abgefangen.

Der Zusammenhang zwischen den Fehlern und den zurückgegebenen Fehlermeldungen ist in Tab. 3 dargestellt. Damit die Tabelle nicht zu unübersichtlich wird, fehlen die GEMDOS-Funktionen für zeichen-orientierte Ein-/Ausgabe (C...) sowie Uhrzeit/Datum (T...), da sie garkeine bzw. die dokumentierten Fehlercodes zurückgeben.

Bei 'Frename' bezieht sich die Zeile 'Q' auf den Pfad der umzubenennenden Datei, 'Z' auf den neuen Pfad. Die Funktionen 'Fsetdta' und 'Fdatime' liefern keinen definierten Wert zurück ('void'-Funktionen).

Erschrecken Sie bei dieser Tabelle nicht zu sehr, in der Praxis sieht die Sache einfacher aus, da einige Fehler normalerweise nicht auftreten. Trotzdem sollte für sie irgendeine Fehlerbehandlung vorgesehen sein, damit Ihr Programm wenigstens nicht abstürzt.

Für Vollständigkeit kann bei dieser Aufstellung nicht garantiert werden, da man leicht irgendeine Wechselwirkung übersieht. Vielleicht ergeben sich noch im Laufe dieser Serie Änderungen; auch würden mich hier Ihre Erfahrungen interessieren.

In den Spalten sind die bei den intern tatsächlich auftretenden Fehlern gelieferten Fehlercodes eingetragen. Fehlercodes, die wirklich nur unter ganz eindeutigen Umständen eintreten können, sind in der letzten Spalte gesondert aufgeführt.

Außer den symbolischen Bezeichnungen finden sich in der Tabelle manchmal folgende Einträge:

"Bomben"

Ein Fehler kann zwar auftreten, wird aber von GEMDOS nicht erkannt. Da GEMDOS mit falschen Werten weiterarbeitet (mit Vorliebe Null-Zeiger), gibt es sofort Bomben. Immerhin weiß man dann, daß etwas schiefgegangen ist.

Dies heißt jedoch keineswegs, daß hiermit alle Fälle abgedeckt sind, bei denen es bombt, da in der Tabelle nur die unmittelbar auftretenden Fehler aufgeführt sind. Durch geschickte Kombinationen von GEMDOS-Aufrufen und Fehlern im GEMDOS lassen sich ebenfalls Bomben erzeugen.

"unbem."

Ähnlich wie "Bomben", nur daß das Nicht-Bemerken des Fehlers nicht unmittelbar zu unangenehmen Konsequenzen führt. Meistens passiert nicht viel, aber man sollte sich nicht über Spätfolgen in Form merkwürdiger Nebeneffekte wundern.

"k. Fehler"

Der Fehler wird zwar bemerkt und richtig behandelt, aber es erfolgt keine Fehlermeldung über die üblichen Fehlercodes. Der Fehler kann vom Anwenderprogramm höchstens indirekt erkannt werden.

"0L"

'Malloc' liefert im Gegensatz zu den anderen Funktionen Null als Fehler, dies entspricht aber der Dokumentation.

Doch nun zur Beschreibung der Fehlerursachen (Spalten der Tab.3 von links nach rechts):

"Kein freier interner Speicher mehr"

Der von GEMDOS für die Verwaltung von Laufwerken, Directories und Dateien benötigte interne Speicher reicht nicht mehr aus. Dieser Fehler wird zwar fast überall abgefangen, aber da er zu den unterschiedlichsten Fehlermeldungen führt, und der Desktop ihn nicht erkennt bzw. erkennen kann, sollte man dafür sorgen, stets ausreichend internen Speicher zur Verfügung zu haben (vor allem wichtig bei Harddisk-Besitzern). Genauere Erläuterungen zum internen Speicher s. "Speicherverwaltung".

"Laufwerk unbekannt"

Das angesprochene Laufwerk ist GEMDOS unbekannt und nicht vorhanden. Ein Laufwerk ist bekannt, sobald es zum ersten Mal angesprochen wurde und vorhanden war. Es bleibt bekannt, auch wenn es vielleicht später nicht mehr vorhanden sein sollte. In diesem Fall wird dies nicht vor Ausführung der GEMDOS-Funktion erkannt, sondern bestenfalls beim eigentlichen Zugriff über das BIOS.

GEMDOS geht halt davon aus, daß Laufwerke im System bleiben, wenn sie erst einmal installiert sind, was normalerweise ja auch zutrifft. Diese Zusammenhänge werden noch in der späteren Folge "Disk-Verwaltung" erklärt.

"Illegaler Funktions-Modus"

Der der GEMDOS-Funktion übergebene Modus-Parameter (abhängig von der Funktion) liegt nicht im vorgesehenen Werte-Bereich (z.B. darf 'mode' bei 'Fseek' nur die Werte 0,1 oder 2 haben). Dies wird bei vielen Funktionen nicht bemerkt, hat aber (meistens) keine schlimmen Nachwirkungen.

"Kein Standard-Handle"

Die GEMDOS-Funktion erwartet ein Standard-Handle (0...5). Es werden hier tatsächlich alle illegalen Handles abgefangen.

"Illegales Datei-Handle"

Das übergebene Handle ist kein Datei-Handle (6...74) bzw. das Device- oder Standard-Handle eines auf eine Datei umgeleiteten Devices.

	kein Speicher mehr	kein interner Speicher mehr	unbek. Lauf-werk	illegaler Funkt.-Modus	kein Stand.-Handle	illegales Datei-Handle	k. Datei-Handles mehr	k. Pfad-Handles mehr	Datei nicht gefund.	Pfad nicht gefund.	'.' o. '..' als Name	Nur-Lesen-Datei	(auch) anderer Prozeß	Diskette voll	Root-Direct. voll	illegaler Subdir.-Zugriff	interner Fehler	BIOS	spezielle Fehler
Dcreate		EPTHNF, ENSMEM	EPTHNF				ENHNDL	EPTHNF		EPTHNF	EPTHNF	EACCDN		EACCDN	EACCDN	EACCDN		×	
Ddelete		EPTHNF, ENSMEM	EPTHNF					EPTHNF		EPTHNF		unbem.				EACCDN	EINTRN	×	
Dfree		ERROR	ERROR					ERROR										×	
Dgetdrv																			
Dgetpath		EDRIVE	EDRIVE					EDRIVE											
Dsetdrv			unbem.																
Dsetpath		ENSMEM	ERROR					ERROR										×	
		EPTHNF						EPTHNF		EPTHNF									
Fattrib		EPTHNF	EPTHNF	unbem.				EPTHNF	EFILNF	EPTHNF	EFILNF					EFILNF		×	
Fclose						EIHNDL							unbem.				EINTRN	×	
Fcreate		EPTHNF, ENSMEM	EPTHNF	unbem.			ENHNDL	EPTHNF		EPTHNF	EPTHNF	EACCDN	unbem.	EACCDN	EACCDN	EACCDN		×	
Fdatime						Bomben							unbem.					×	
Fdelete		EFILNF	EFILNF					EFILNF	EFILNF	EFILNF	EFILNF	EACCDN	EACCDN			EFILNF		×	
Fdup					EIHNDL		ENHNDL												
Fforce					EIHNDL	EIHNDL													
Fgetdta																			
Fopen		EFILNF, ENSMEM	EFILNF	unbem.			ENHNDL	EFILNF	EFILNF	EFILNF	EFILNF	EACCDN				EFILNF		×	
Fread						EIHNDL							unbem.					×	
Frename		ENSMEM																	ENSAME
(Q)		EPTHNF, EFILNF	EPTHNF				ENHNDL	EPTHNF	EFILNF	EPTHNF	EFILNF	EACCDN	unbem.			EFILNF	EINTRN	×	
(Z)		EPTHNF	EPTHNF				Bomben	EPTHNF	(1)	EPTHNF	unbem.		unbem.	Bomben	Bomben		k.Fehler	×	
Fseek				EINVFN		EIHNDL												×	ERANGE
Fsetdta																			
Fsfirst		EFILNF	EFILNF	unbem.				EFILNF	EFILNF	EFILNF								×	
Fsnext		ENMFIL							ENMFIL									×	
Fwrite						EIHNDL						(2)	unbem.	k.Fehler				×	
Malloc	0L	0L																	
Mfree													unbem.						EIMBA
Mshrink		Bomben											unbem.						EIMBA, EGSBF
Pexec	ENSMEM	EFILNF, ENSMEM	EFILNF	EINVFN			ENHNDL	EFILNF	EFILNF	EFILNF	EFILNF	(3)				EFILNF	k.Fehler	×	EPLFMT
Pterm																			
Pterm0																			
Ptermres		Bomben																	

(Q) alter Pfad (Z) neuer Pfad (1) EACCDN falls Name schon existiert (2) unbem. falls 'Nur-Lesen' nach Fopen gesetzt (3) manchmal EACCDN wegen Fehler im GEMDOS

Tab. 3 - Fehlermeldungen der GEMDOS-Funktionen

"Kein freier Benutzer-Speicher mehr"

Der größte, zusammenhängende freie Speicherbereich reicht nicht mehr für das gewünschte 'Malloc' oder für das Laden bzw. Starten eines Tochterprozesses aus.

Achtung: Es wird nur überprüft, ob zu dem Datei-Handle eine geöffnete Datei existiert. Wenn das Handle nicht in dem für Handles allgemein zulässigen Wertebereich von -3 bis + 74 liegt, können vollkommen unvorhersehbare Effekte auftreten, da GEMDOS auf falsche Speicherbereiche zugreift (Array-Index-Überlauf)!

Daher ist es besonders wichtig, eine Fehlerbehandlung bei 'Fopen'/'Fcreate' durchzuführen, da ansonsten der Fehlercode (i.a. kleiner als -32!) als Datei-Handle weiterverwendet wird.

"Keine Datei-Handles mehr"

Es ist kein Datei-Handle mehr frei. Bei 69 möglichen Datei-Handles sollte dies eigentlich nie der Fall sein, doch gibt es Situationen, bei denen GEMDOS fehlerhafterweise Datei-Handles verbraucht, so daß man nicht sicher kann, daß GEMDOS nicht doch irgendwann einmal – auf Grund eines noch nicht entdeckten Fehlers vielleicht – alle Handles aufbraucht.

"Keine Pfad-Handles mehr"

Es sind keine Pfad-Handles mehr frei. GEMDOS braucht zur Verwaltung der Standard-Pfade von Laufwerken interne Datenstrukturen, die über (nur intern bekannte) Handles angesprochen werden. Näheres hierzu später in der Folge über "Directory-Verwaltung". Die Maximalzahl von 39 Pfaden wird in der Praxis wohl kaum erreicht werden, so daß dieser Fehler nicht zu befürchten ist.

"Datei nicht gefunden"

Ein der GEMDOS-Funktion übergebener Dateiname wurde im angesprochenen Directory nicht gefunden.

"Pfad nicht gefunden"

Ein Pfad für eine Datei wurde nicht gefunden.

" '.' oder '..' als Dateiname"

Der Pseudo-Directory-Name '.' oder '..' wurde als Dateiname verwendet, wo es nicht zulässig ist.

"Nur-Lese-Datei"

Es wurde ein Schreibzugriff auf eine Datei versucht, die mit dem Nur-Lesen-Attribut versehen ist. Dazu zählt auch das versuchte Löschen der Datei, nicht jedoch das Ändern von DirectoryInformationen (Name, Attribut, Datum/Zeit).

'Pexec' sollte selbstverständlich immer funktionieren, aber beim Aufruf von 'Fopen' innerhalb der 'Pexec'-Funktion wurde die Übergabe des Modus-Wortes vergessen. Befindet sich auf dem Stack zufällig ein Modus-Wort ungleich Null, so daß versucht wird, die Datei (auch) zum Schreiben zu öffnen, klappt dies natürlich nicht bei gesetztem 'Nur-Lesen'-Attribut. Beim Start vom Desktop aus trifft dies anscheinend nicht zu, aber bei Shell-Programmen kann man sich da ja nicht sicher sein. Außerdem werden nicht zulässige Modi beim 'Fopen' auch nicht abgefangen, allerdings ohne Schaden anzurichten. Mit anderen Worten: Daß Ihr ST überhaupt Programme lädt, ist purer Zufall.

"Datei/Speicher gehört (auch) noch zu anderem Prozeß"

Ein unzulässiger Zugriff auf eine Datei, die von mehr als einem Prozeß geöffnet wurde, wurde versucht. Zur Multi-Tasking-Fähigkeit der Dateiverwaltung s. Folge "Dateiverwaltung". Doch schon vorweg: Hier funktioniert so manches nicht, daher ist diese Fehlermeldung mit Vorsicht zu genießen.

Achtung: Es ist ohne weiteres möglich, mit Datei-Handles fremder Prozesse auf deren Dateien zuzugreifen.

"Diskette voll"

Dies wird günstigstenfalls bei Directory-Operationen gemeldet. Beim Datei-Zugriff ('Fwrite') muß anhand der zurückgelieferten Anzahl der tatsächlich geschriebenen Zeichen selbst festgestellt werden, ob die Datei korrekt beschrieben wurde.

"Root-Directory voll"

Das Root-Directory (Haupt-Directory, oberste Ebene) ist voll. Beim Standard-Disketten-Format sind 112 Einträge möglich, so daß es hier keine Probleme geben sollte. Die Größe von Subdirectories (Ordnern) ist nur durch die Disketten-Kapazität begrenzt.

"Illegaler Subdirectory-Zugriff"

Es wurde versucht, ein nichtleeres Subdirectory zu löschen ('Ddelete'), eine mit einem Subdirectory namensgleiche Datei zu erzeugen ('Fcreate'), ein schon vorhandenes Subdirectory erneut zu erzeugen ('Dcreate') oder ein Subdirectory mit nur für Dateien vorgesehenen Funktionen zu bearbeiten.

"Interner Fehler"

Dieser Fehler erfolgt, wenn GEMDOS-eigene Datenstrukturen in sich widersprüchlich sind. Dieser Fehler kann also nur auftreten, wenn zuvor schon Fehler im GEMDOS ihr Unwesen getrieben haben.

Solche Abfragen auf interne Fehler werden leider nur bei 'Fclose' und 'Ddelete' gemacht, obwohl sie sicherlich auch an anderen Stellen wünschenswert wären, um Schlimmeres zu verhindern. Denn GEMDOS ist – wie auch das GEM – sehr empfindlich gegenüber falschen Parametern usw., da kaum Kontrollen gemacht werden. Kleinste Programmfehler in Anwenderprogrammen können so GEMDOS ziemlich verwirren. Hinzu kommen noch die Fehler im GEMDOS selbst, die manchmal sehr versteckt sind, aber ganz unerwartet in Erscheinung treten.

"BIOS-Fehler"

In dieser Spalte sind alle GEMDOS-Funktionen markiert, bei denen Disk-Zugriffe auftreten können. Dabei verwendet GEMDOS ausschließlich die BIOS-Routinen 'Rwabs', 'Mediach' und 'Getbpb'. Von den so gekennzeichneten Funktionen können also neben den GEMDOS-Fehlermeldungen auch BIOS-Fehlermeldungen zurückgegeben werden.

Zur Diskettenfehler-Behandlung habe ich ja schon in der letzten Folge einiges gesagt, aber eines bleibt noch nachzutragen. Da die GEMDOS-Funk-

tion dann einfach abgebrochen wird, kann es sein, daß interne Datenstrukturen sich noch in einem undefinierten Zustand befinden, weil Disk-Zugriffe manchmal während der Manipulation dieser stattfinden. Normalerweise wirkt sich dies – wie die Erfahrung zeigt – nicht negativ aus, aber die Folgen daraus sind schwer abzusehen, so daß ich nur dazu raten kann, fehlerfreie Disketten zu verwenden. Noch gefährlicher ist der Diskettenwechsel während einer Disk-Operation, da dann das GEMDOS-Kommando einfach wiederholt wird, was noch undurchschaubarere Folgen hat.

Zum Schluß kommen wir zu den Fehlermeldungen, die nur bei ganz bestimmten GEMDOS-Funktionen auftreten können und eine eindeutige Bedeutung haben.

ENSAME

Die Funktion 'Frename' kann Dateien nur innerhalb eines Laufwerks umbenennen. Bei einem anderen Laufwerk mit anderer Kennung müßte die Datei ja kopiert werden. Da GEMDOS über keine Kopierfunktion verfügt, wird mit dieser Fehlermeldung abgebrochen.

ERANGE

Mit 'Fseek' wurde versucht, den Datei-Positions-Zeiger auf einen illegalen Wert zu setzen. Negative Werte und Werte größer als die Datei-Länge sind nicht zugelassen. Dies gilt auch bei einer zum Schreiben geöffneten Datei, bei der der Zeiger nur max. auf das Dateiende gesetzt werden darf.

EIMBA

Die an 'Mfree' und 'Mshrink' übergebene Adresse ist nicht die eines mit 'Malloc' reservierten Speicherbereichs. Es wird nicht kontrolliert, ob der Speicherbereich überhaupt von dem eigenen Prozeß reserviert wurde, d.h. man kann auch fremde Bereiche freigeben bzw. verkleinern (sehr gefährlich!).

EGSBF

Es wurde versucht, mit 'Mshrink' einen Speicherbereich zu vergrößern. Die Speicherverwaltung des GEMDOS ist dazu jedoch nicht in der Lage, die neue Länge des Bereichs muß kleiner gleich der alten sein.

EPLFMT

'Pexec' hat versucht, eine Datei zu laden, deren Format nicht dem einer Programm-Datei entspricht. Das Format wird als illegal betrachtet, wenn das erste Wort des Programm-Headers nicht $601A ist, oder wenn die Relozier-Offsets aus dem zulässigen Bereich (TEXT- und DATA-Segment) herausführen.

In der nächsten Ausgabe werden wir uns dann dem Critical-Error-Handler und der Form-error-Routine widmen.

Alex Esser



MORTVILLE MANOR

MORTVILLE MANOR ist ein Spiel, bei dem Sie in die Rolle eines Detektivs schlüpfen. Sie sitzen gelangweilt in ihrem Büro, als plötzlich ein Brief von ihrer früheren Freundin Julia eintrifft. In dem Brief steht: Sehr krank – hoffnungslose Situation – Todesgefahr – sehr beunruhigt wegen meiner Familie – hoffe schnellstens auf ihre Hilfe – Diskretion erforderlich – was immer auch geschieht auf der Mauer des Schweigens, ich hinterlasse ihnen eine Nachricht.

Sie machen sich natürlich sofort auf den Weg. Doch dort angekommen wird ihnen gesagt, daß Julia gestorben sei. Für Sie beginnt nun die Jagd nach dem vermeintlichen Mörder.
Zu Beginn des Spiels fällt sofort die Sprachausgabe auf; auch die Personen auf Mortville Manor, die Sie befragen, 'sprechen' mit Ihnen. In der Menüleiste können Sie eine der acht Personen anwählen, falls sie anwesend ist. Dann erscheint ein Fragenkatalog mit Fragen über Julia, über die Beziehungen der Personen untereinander, die Erbschaft und einiges mehr. Die Antworten sollte man sich irgendwo aufschreiben, um dann in Ruhe kombinieren zu können.

Aber was wäre ein Detektiv, ohne zu schnüffeln? In einer anderen Menüleiste sind Wörter wie zum Beispiel 'öffnen' oder 'untersuchen'. Mit Hilfe dieser Wörter kann man ganze Zimmer untersuchen; aber seien Sie vorsichtig, daß Sie keiner dabei beobachtet. Welche Personen anwesend sind, kann man immer über eine Anzeige erfahren.

MORTVILLE MANOR ist eine gut gemachte Detektiv-Story. Die Sprachausgabe sei nochmals erwähnt. Deshalb sollte man beim Kauf auch unbedingt darauf achten, daß man eine deutsche Version bekommt, wenn man des Englischen nicht mächtig ist.

AS

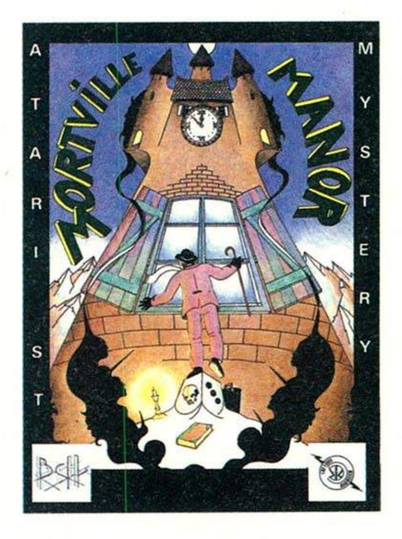
MORTVILLE MANOR

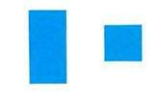
Kategorie: **Detektivspiel**
Besonderheiten: **Sprachausgabe**

Spieler: **1**
Monitor: **color/monochrom**
Steuerung:
Hersteller: **Kyilkhor**
Vertrieb:

Preis:

Wertung:





STAR TREK

Wer kennt sie nicht – die Geschichten von STAR TREK, die vielleicht besser unter dem Seriennamen 'Raumschiff ENTERPRISE' bekannt wurden?

Zunächst eine kleine Vorgeschichte: Die Klingonen haben entdeckt, daß ein merkwürdiger Stoff, der nur auf dem Planeten Dekian II existiert, telepathische Kräfte entwickelt. Da der Stoff nur auf kurze Distanz oder mit Hilfe einer Energiequelle funktioniert, haben sich die Kingonen einen Trick ausgedacht. Sie nähern sich einen Schiff der Föderation und täuschen einen Waffenstillstand vor. Nachdem sie die telepathische Oberherrschaft über das Schiff gewonnen haben, brauchen sie nur eine kleine Ladung des Stoffes in den Antrieb des Schiffes einzubauen, und ihre Kontrolle über die Besatzung ist gesichert. Nachdem die Vereinigten Föderierten Planeten schon zwanzig Raumschiffe auf diese Art verloren haben, sieht sich das Flottenkommando gezwungen, eine Barriere um das rebellierende Gebiet zu legen, das von keinem materiellen Objekt durchdrungen werden kann. Doch bevor die Barriere aufgebaut wird, fliegt die ENTERPRISE in das Gebiet, mit der Aufgabe, die Schäden innerhalb von fünf Jahren zu beheben. Schaffen sie es nicht, sind sie und die Planeten für immer abgeschottet.

Bei Beginn des Spiels befinden sich auf dem Bildschirm ein großes Bild (Primärbereich) und sieben kleine. Auf dem Primärbereich ist die Brükke der Enterprise mit den sieben Offizieren zu sehen, auf den kleinen Bildern die jeweiligen Köpfe. Die Offiziere haben unterschiedliche Aufgabenbereiche, die man durch Anklicken der Köpfe mit Hilfe der Maus oder einem Joystick erreichen kann. Klickt man Captain Kirk an, kommt man an die Funktionen: Laden, Sichern, Pause, abgelaufene Zeit und eliminierte Feinde; Spock: Sonnensystem- und Planetenerkennung, Feindstatus und Zustand der Enterprise; Sulu: Sternenglobus, Antriebskontrolle und Sonnensystem; Scotty: Warp- und Impulsstatus; Chekov: Bewaffnung, Ortung und Visiereinrichtungen; McCoy: Gesundheitszustand der Besatzung; Uhura: Kommunikation.

Um nun ein Sonnensystem anzufliegen, muß zuerst der Kurs festgelegt werden. Dazu wird Sulu angeklickt und der Sternenglobus in den Primärbereich gebracht. Nach der Wahl des Sonnensystems wird die Antriebssteuerung in den Primärbereich gebracht und die Geschwindigkeit eingestellt. Geschwindigkeiten von einem bis zu zehn Warp (Warp=Antrieb der Enterprise) sind möglich, Sie müssen jedoch bedenken, daß Sie nicht unbegrenzt Energie an Bord haben und daher reicht es mit höchstens acht Warp zu fliegen.

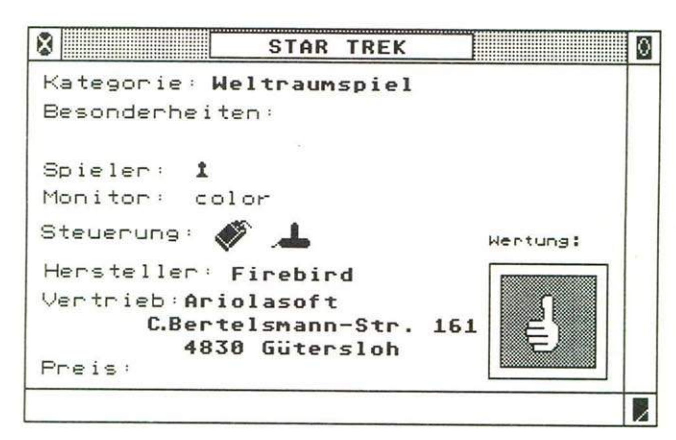


Ertönt nach kurzer Zeit ein Glockensignal, bedeutet das, daß Sie das Ziel erreicht haben. Auf dem Bildschirm ist nun das Sonnensystem zu sehen. Klickt man nun einen Planeten an und fragt dann Spock danach, wird er Ihnen sagen, daß er zum Beispiel bewohnbar ist. Bestätigen Sie, indem Sie das Fenster 'Confirm' anklikken und auf volle Impulsleistung gehen, dann nähert sich die Enterprise dem Planeten. Dort angekommen, besteht nun die Möglichkeit, sechs Personen auf den Planeten 'hinunterzubeamen'. Auf dem Planeten können Sie wichtige Dinge finden, die Sie zur Lösung Ihrer Aufgabe brauchen, Sie können aber auch von feindlichen Lebewesen angegriffen werden. Nicht nur auf den Planeten, auch im All kann es zu einem Angriff kommen.

Um das Spiel erfolgreich zu beenden, gibt es mehrere Möglichkeiten. Sie können zum Beispiel den Herd des Übels, den Planeten Dekian II, zerstören, oder ganz einfach den Klingonenadmiral gefangennehmen, der sich im Hauptquartier auf einem der Klingonenplaneten befindet.

STAR TREK ist ein tolles Spiel, das keinerlei Langeweile aufkommen läßt. Gute Grafik, guter Sound, gute Idee und ein guter Aufbau lassen in diesem Spiel nichts vermissen.

AS



STAR TREK

Kategorie: **Weltraumspiel**
Besonderheiten:

Spieler: 1
Monitor: color
Steuerung:
Hersteller: **Firebird**
Vertrieb: **Ariolasoft**
C.Bertelsmann-Str. 161
4830 Gütersloh
Preis:

Wertung:

Warum AT

DATA BECKER

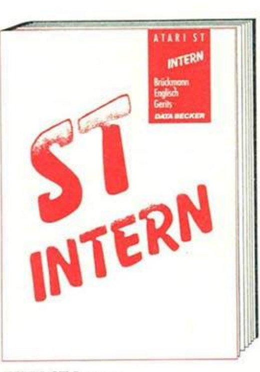
Atemzug gen

1. Die Standardwerke

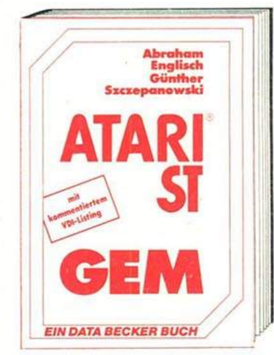
Beispielhaft für unsere Standardwerke sei hier unser ST-Intern-Band genannt. In der jetzt völlig überarbeiteten Neuauflage noch besser strukturiert und erstmalig mit einer ausführlichen Blitter-Dokumentation. Unentbehrlich für jeden engagierten ST-Anwender Ein Standardwerk eben.



ATARI ST für Einsteiger
248 Seiten, DM 29,–



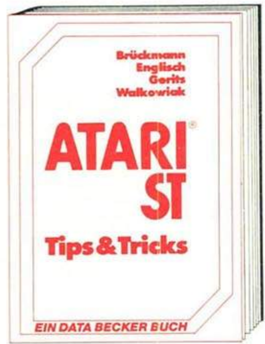
ATARI ST Intern
Hardcover, 637 Seiten, DM 69,–



ATARI ST GEM
Hardcover, 691 Seiten, DM 69,–

2. Die ST-Bibliothek

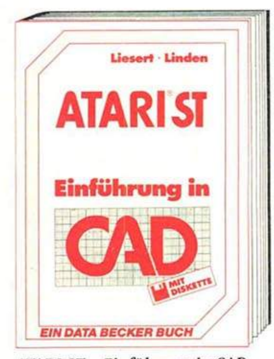
Ob frischgebackener ST-Besitzer oder ambitionierter 68000er-Programmierer – wenn Sie Ihren ATARI ST effizient und professionell einsetzen wollen, brauchen Sie hochkarätige Informationen von kompetenten Autoren. Informationen, die Sie in der „ST-Bibliothek" von DATA BECKER finden können.



ATARI ST Tips & Tricks
352 Seiten, DM 49,–



C für Einsteiger
393 Seiten, DM 39,–



ATARI ST – Einführung in CAD
Hardcover, 289 Seiten
inkl. Diskette, DM 69,–
GFA 2.0 erforderlich

3. Unsere Aktuellsten

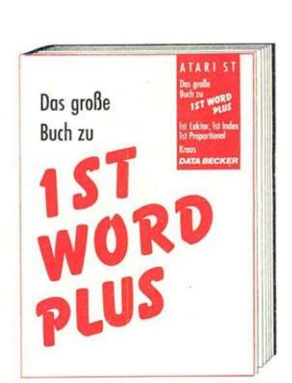
Der ATARI ST hat sich inzwischen zum eigenen Standard voll etabliert. Doch die Zeit bleibt nicht stehen. Wir, die wir von Anfang die Entwicklung des ST mit aktuellen, intelligenten Sachbüchern begleiten, werden auch weiterhin die Zeichen der Zeit erkennen und dem Anwender die Literatur bieten, die er braucht.



Das große Buch zum MEGA ST
ca. 400 Seiten, DM 49,–
erscheint ca. 1/88

Alles zum neuen Super-ST: Einstieg, DTP, Arbeiten mit dem Laserdrucker, Software, TOS und Blitter. Zu finden im großen Buch zum MEGA ST.

Das große Buch zu 1st Word Plus – endlich mit allen Informationen zu 1st Word Plus und den Zusatzprodukten 1st Lektor, 1st Proportional und 1st Index.



Das große Buch zu 1st Word Plus
Hardcover, ca. 300 Seiten
inkl. Diskette, DM 59,–
erscheint ca. 12/87

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

ARI ST und meist im gleichen annt werden.



ATARI ST Floppy und Harddisk
Hardcover, 522 Seiten, DM 59,–

4. Alles zum Thema Grafik

Setzen Sie die Grafikfähigkeiten Ihres ST gezielt für Ihre eigenen Anwendungen ein. Diese Bücher zeigen Ihnen, was alles möglich ist. Von einer flackerfreien Animation bis hin zu atemberaubenden 3-D-Grafiken finden Sie hier das gesamte Know-how zum Thema Grafik.



Das Supergrafikbuch zum ATARI ST
Hardcover, 838 Seiten,
inkl. Diskette, DM 69,–



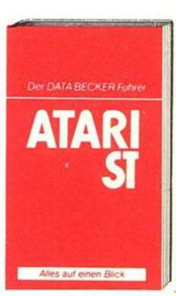
3-D-Programmierung
Hardcover, 601 Seiten,
inkl. Diskette, DM 69,–



as Maschinensprachebuch zum
TARI ST
34 Seiten, DM 39,–

5. Die DATA BECKER Führer

Kompakte Informationsquellen, die den Anwender bei seiner Arbeit mit dem ST nicht allein lassen. Auf einen Blick findet er alle wichtigen Kommandos und Befehle. Schnell und zuverlässig. Für die tägliche Arbeit am Rechner einfach unentbehrlich.



Der DATA BECKER Führer zum ATARI ST
240 Seiten, DM 29,80



Der DATA BECKER Führer zu GFA-BASIC
254 Seiten, DM 24,80



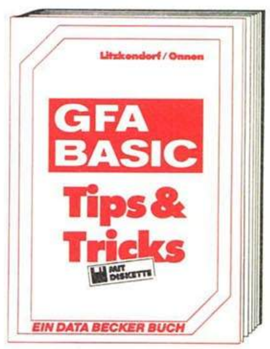
Der DATA BECKER Führer zu 1st Word
192 Seiten, DM 24,80

6. Die GFA-Bücher

Vom Einstieg bis hin zu all den raffinierten Tricks echter Profis – drei Bücher zum GFA-BASIC sorgen dafür, aß Sie die fantastischen Möglichkeiten ieser wohl leistungsstärksten BASIC-ersion auch wirklich alle für Ihre igenen Programme voll ausschöpfen önnen.



Das große GFA-BASIC-Buch
Hardcover, 574 Seiten, DM 49,–



GFA-BASIC Tips & Tricks
Hardcover, 350 Seiten,
inkl. Diskette, DM 49,–

COUPON

An: DATA BECKER · Merowingerstr. 30
4000 Düsseldorf
Bitte senden Sie mir:

____

____

zzgl. DM 5,– Versandkosten
unabhängig von der bestellten Stückzahl
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name ______________________

Straße ______________________

Ort ______________________

Terrorpods

Nach dem erfolgreichen Barbarian veröffentlicht Psygnosis jetzt Terrorpods. Es ist nicht, wie erwartet, ein Abklatsch von Barbarian geworden, denn Terrorpods liegt eine völlig andere Idee zugrunde. Das Geschehen des Aktion-Strategie-Spiels spielt sich auf dem fernen Planeten Colian ab, große Temperaturunterschiede machen das Wohnen dort ungemütlich, aber wegen des enormen Reichtums an Bodenschätzen ist der Planet von unschätzbarem Wert. Dementsprechend sind die Industrieanlagen zahlreich, neben Minen existieren Aufbereitungsanlagen, Warenlager, Pumpstationen und etliches mehr. Die Aufgabe des Spielers besteht nun darin, die feindlich gesinnten Terrorpods zu eliminieren, welche leider die Industrieanlagen zerstören. Doch das ist leichter gesagt als getan, die Feuerkraft des Fahrzeugs, mit dem Sie über den Planeten streifen, ist begrenzt, ebenfalls die Brennelemente, die das Gefährt vorwärts bewegen. Außerdem genügt eine Salve nicht, um die Terrorpods zu vernichten, man vertreibt sie nur. Eine stärkere Waffe muß dazu herangezogen werden, doch die benötigt viel Energie. Während man sich die unliebsamen Terrorpods vom Leibe hält, müssen immer wieder Warenlager, Fabriken und Tankstellen angefahren werden, um seinen Energiehaushalt aufzufrischen. Dazu dient übrigens ein kleines Roboterfahrzeug, das vom Mutterschiff ausgesetzt wird, um die verschiedenen Stützpunkte anzulaufen.





Nicht nur die Terrorpods werden einem gefährlich, sondern auch ab und zu auftauchende Torpedos, denen man wegen der Energieersparnis ausweichen sollte, was übrigens sehr leicht geht.



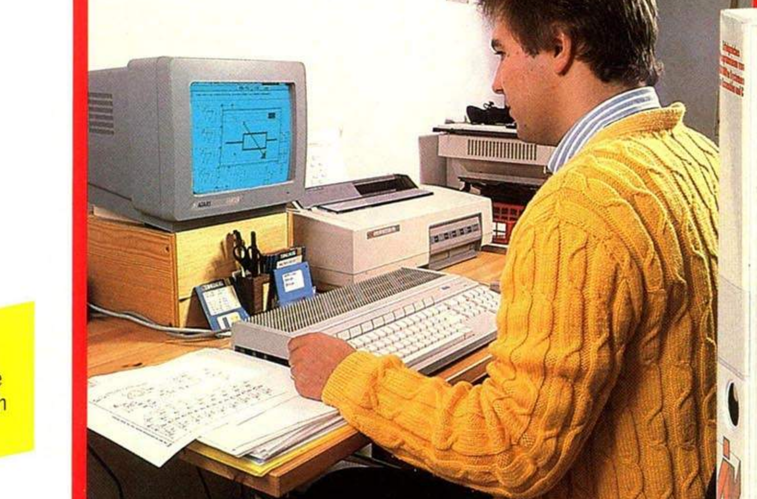
Terrorpods besticht durch die Vielzahl der Möglichkeiten. Eine Karte kann zur besseren Orientierung herangezogen oder zerstörte Anlagen wieder aufgebaut werden. Doch damit nicht genug, neben dem komplexen Spielverlauf stimmt auch das Drumherum. Das Spiel kann abgespeichert und zu einem späteren Zeitpunkt fortgesetzt werden, ebenso kann beim Schwierigkeitsgrad zwischen sechs Stufen gewählt werden. Der Highscore wird natürlich auch abgespeichert. Zu Beginn des Spiels kann man unter verschiedenen Nationalitäten wählen, dementsprechend sind auch die Kommentare während des Spiels in der jeweiligen Landessprache. Die deutschen Kommentare sind zwar zum Teil etwas holprig, was aber nicht sonderlich stört. Das Scrolling ist gut, bedenkt man, daß der Hintergrund langsamer scrollt als der Vordergrund.

Zum Lieferumfang gehört neben der Diskette ein Anleitungsbogen und ein Roger Dean-Poster, welche die Terrorpods in Aktion zeigt.

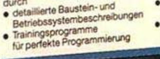
Fazit:

Wer nicht nur ballern will, sondern auch Strategieelemente liebt, dem kann zu diesem Spiel nur geraten werden. Die ausgezeichnete Grafik sowie die Spielmotivation sprechen nur für das Spiel. Psygnosis hat mit Terrorpods wieder einen Hit gelandet, der sicher seinen Weg in der ST-Welt machen wird.

68000er-Systeme perfekt programmieren in Assembler und C



Erfolgreiches Programmieren von 68000er Systemen in Assembler und C





Erst prüfen, dann kaufen
Schauen Sie sich dieses Werk in Ruhe an: 10 Tage lang dürfen Sie Ihr Ansichtsexemplar unverbindlich zu Hause prüfen.

Nutzen Sie die Stärken Ihres Atari!
Jetzt hilft Ihnen dieses neue Nachschlagewerk,
■ effiziente Problemlösungen auf **Betriebssystemebene** zu realisieren
■ anspruchsvolle **Anwenderprogramme** zu entwickeln
■ mit ausgefeilter **Bausteinprogrammierung** eine neue Atari-Dimension zu erschließen.

Die wichtigsten Themen auf einen Blick:
■ **detaillierte Hardware-Beschreibungen** der Prozessorfamilie 68000 und ihrer 8-, 16-, 32-Bit-Peripheriebausteine mit technischen Kennwerten, Pinbelegung, internen Architekturen, Befehlsbeschreibungen;
■ **Betriebssysteme:** Sie lernen die Strukturen der 68000er-Betriebssysteme (wie GEM-TOS, OS-9 und Unix) kennen und erfahren, wie Sie die Systemroutinen zur Optimierung Ihrer Assembler- und C-Programme nutzen;
■ **Softwareengineering:** die optimale Vorgehensweise von der Problemanalyse über die Codierung bis hin zur abschließenden Dokumentation;
■ **Programmierkurse für effiziente Anwender-, System- und Bausteinprogrammierung in Assembler und C.** Damit verfügen Sie zugleich über sofort einsetzbare Routinen, z. B. zur Drucker- und Bildschirmansteuerung;
■ **die C-Compiler-Werkstatt:** Schritt für Schritt programmieren Sie selbst einen C-Compiler inkl. Funktionsbibliotheken;
■ **Assembler-Makrobibliotheken, Anwender- und Funktionsbibliotheken in C;**
■ **Tips und Utilities** wie Schnittstellen- und Backup-Routinen, Fensterverwaltung;
■ **bewährte Musterlösungen** wie zu Rechnerkopplung und Multitasking;
■ **Anleitung für raffiniertes Systemtuning,** z. B. durch Einsatz schnellerer Peripheriebausteine/Höhertakten der CPU;
■ **Praxiswissen zu speziellen Einsatzbereichen** wie Messen, Steuern, Regeln;

Dieses Werk veraltet nie
Unsere 68000er-Fachredaktion versorgt Sie regelmäßig und zuverlässig mit aktuellen Informationen über Hardware- und Betriebssystem-Weiterentwicklungen, weiteren Utilities und Musterlösungen sowie Bibliothekserweiterungen in Assembler und C.

Fordern Sie noch heute mit nebenstehender Bestellkarte an:

Erfolgreiches Programmieren von 68000er-Systemen in Assembler und C

stabiler Ringbuchordner, Format DIN A 4, Grundwerk ca. 400 Seiten. Bestell-Nr. 3400, Preis: DM 92,– (erscheint ca. 1. Quartal 1988)

Alle 2–3 Monate erhalten Sie Ergänzungsausgaben zum Grundwerk mit je ca. 120 Seiten zum Seitenpreis von 38 Pfennig (Abbestellung jederzeit möglich).



INTEREST-VERLAG
Fachverlag für anspruchsvolle Freizeitgestaltung

Industriestraße 21
D-8901 Kissing
Tel. 08233/23900

Bitte abtrennen

Ja, senden Sie mir bitte sofort

☐ Expl. **„Erfolgreiches Programmieren von 68000er-Systemen in Assembler und C“**

stabiler Ringbuchordner, Format DIN A 4, Grundwerk ca. 400 Seiten, Bestell-Nr. 3400, Preis: DM 92,– (erscheint ca. 1. Quartal 1988)

Alle 2–3 Monate erhalten Sie Ergänzungsausgaben zum Grundwerk mit je ca. 120 Seiten zum Seitenpreis von 38 Pfennig (Abbestellung jederzeit möglich).



Meine Anschrift:

Name

Vorname

Straße, Haus-Nr.

PLZ, Ort

Unterschreiben Sie hier bitte Ihre Bestellung!
Bei Minderjährigen ist die Unterschrift eines gesetzlichen Vertreters erforderlich. Ohne Ihre Unterschrift kann die Ansichtsbestellung nicht bearbeitet werden.

X

Datum Unterschrift

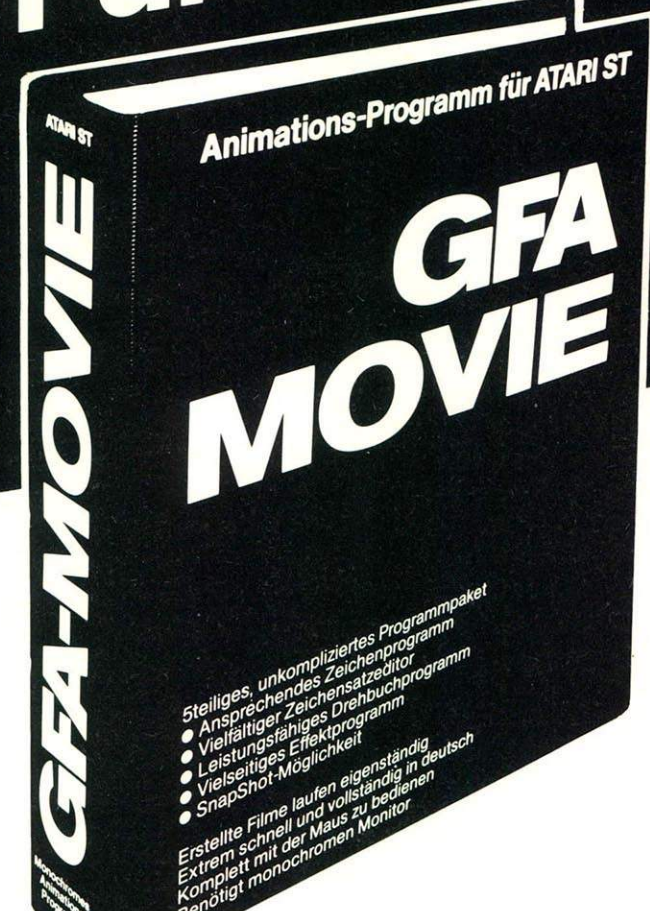
Bitte unterschreiben Sie auch Ihre Sicherheitsgarantie!
Mit der Sie folgendes zur Kenntnis nehmen: Sie haben das Recht, Ihr angefordertes Werk innerhalb von 10 Tagen ab Lieferung an die Bestelladresse zurückzusenden, wobei für die Fristwahrung das Datum der Absendung genügt. Sie kommen dadurch von allen Verpflichtungen aus der Bestellung frei.



Datum Unterschrift

Für alle ATARI ST

neu von GFA



GFA-MOVIE DM 149,–

Entschuldigung

Wir haben es uns leichter vorgestellt, ein derart umfangreiches Programmpaket wie den GFA-PUBLISHER in so kurzer Zeit fertigzustellen.

Die Mitte November vorliegende Version genügte unseren Ansprüchen nicht.

Wir sind dennoch von dem Konzept des GFA-PUBLISHER absolut überzeugt – beispielsweise die interne Auflösung von 2540 dpi und die leistungsfähige dreisprachige Silbentrennung sind zukunftsweisend für den gesamten DTP-Markt.

Wir garantieren:

Der GFA-PUBLISHER wird den Leistungsmerkmalen unserer Produktinformation entsprechen … hierzu brauchen wir Zeit.

Zusätzliche Autoren müssen sich in den Code einarbeiten. Routinen müssen korrigiert und optimiert werden. Auch ein ausführlicher Test braucht seine Zeit.

Den endgültigen Auslieferungstermin geben wir in der nächsten Ausgabe bekannt.

ATARI ST

Design-Programm für ATARI ST

GFA ARTIST

GFA-ARTIST

- Alle erdenklichen Gestaltungsmöglichkeiten von Farbgrafik und deren Animation
- Erstellen eines Films durch Zusammensetzen verschiedener Animationen
- Arbeitet mit bis zu 1024 Farben gleichzeitig (auch colorcycling)
- Gemeinsame Darstellung der vollen Farbpalette sowie 80-Zeichen-Text
- Kompatibel zu Standard-Malprogrammen
- Erstellte Filme laufen eigenständig
- Extrem schnell und vollständig in deutsch

Benötigt ATARI ST mit 1 MB ROM + ROM-TOS + Farbmonitor

Design Programm Farbmonitor

GFA-ARTIST DM 149,–

…Anruf genügt: 02 11/58 80 11
GFA-CLUB, GFA-PC-Software bitte Info anfordern

GFA Systemtechnik GmbH

Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 02 11/58 80 11



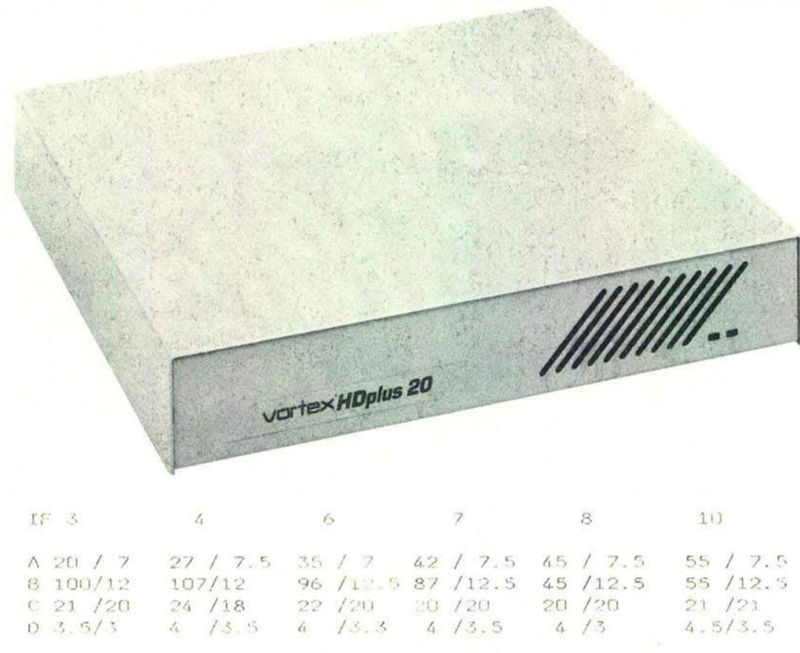
NACH SCHLAG

Die neuen Vortex-Platten im Test

Kaum glaubt sich ATARI mit der SH205 im Vorsprung vor anderen Plattenherstellern, ziehen die auch schon nach. Darunter auch die Firma Vortex, die gleich eine ganze neue Produktreihe vorstellt: Die HDplus-Platten. Im Test mußte sich das kleinste Modell, die HDplus20, bewähren. Muß man in Raunheim jetzt um den Absatz der SH205 bangen? Mehr dazu in diesem Bericht.

Es hat sich einiges getan bei Vortex. Die neuen Festplatten haben 21 bis 128 Megabyte Speicherkapazität und dabei mittlere Zugriffszeiten von 65 Millisekunden (beim kleinsten Modell) bis zu superfixen 28 Millisekunden bei der HD120.

Real programmers don't need manuals, sage ich immer. In diesem Falle aber interessiert mich doch der Vergleich zum alten Handbuch. Fangen wir bei den technischen Daten an. Hier liest man angenehm überrascht, daß



IF	3	4	6	7	8	10
A	20 / 7	27 / 7.5	35 / 7	42 / 7.5	45 / 7.5	55 / 7.5
B	100/12	107/12	96 /12.5	87 /12.5	45 /12.5	55 /12.5
C	21 /20	24 /18	22 /20	20 /20	20 /20	21 /21
D	3.5/3	4 /3.5	4 /3.3	4 /3.5	4 /3	4.5/3.5

(Tabelle 1: Schneller, höher, weiter – die HDplus im Benchmark)

sich nicht nur die Kapazität um ein eher kärgliches halbes Megabyte erweitert hat (auf fast 21MB), sondern daß auch der Datentransfer zwischen Festplatte und ST mit 1.1 Megabyte pro Sekunde abläuft (vorher 937KB pro Sekunde). Ebenso ist die **mittlere Zugriffszeit** von 85 auf 65 Millisekunden gefallen. Darunter versteht man einen rechnerischen Mittelwert für die Positionierung des Lesekopfes einschließlich der Zeit, bis alle störenden Vibrationen abgeklungen sind.

Aufgrund dieser Daten sollte man eigentlich meinen, daß die Platte recht schnell sein könnte. Wie schnell, wird noch nicht verraten, damit die Spannung erhalten bleibt. Nähern wir uns der Platte zunächst in vorsichtigen Umkreisungen.

AUF DEM LAUFSTEG

Von außen sieht sie schon mal richtig deutsch aus, die HDplus. Wie ich das meine? Naja, solides Metallgehäuse im Format des MegaST-Hauptgehäuses, saubere Schnittstellen (DMA IN **und** DMA OUT, gepuffert!) und von außen erreichbare DIP-Schalter zur Umadressierung, damit man auch mehrere Festplatten anschließen kann. Schnörkellos, kühl und sachlich im nicht ganz ATARI-kompatiblen Vortex-Grau. Einziges Zugeständnis ans Freak-Auge: Die im klassischen ST-Winkel schräggestellten Lüftungsschlitze an der Vorderseite – damit kein PC-Besitzer auf Erden glaubt, diese Festplatte sei in erster Linie ihm zugedacht.

Trotzdem, die HDplus-Reihe scheint auch für die riesige Horde der Kompatiblen gebaut zu sein: Der Testplatte lag netterweise ein Backupprogramm bei, sogar mit ausführlicher Anleitung, doch leider auf einer 5.25"-Diskette – für PCs und ATs. Und da ich nicht so recht daran glaube, daß eine 5.25"-Diskette in meine SF314 paßt, selbst wenn ich mich mit der Schere handwerklich an der Diskette vergehe, habe ich dieses Programm leider nicht testen können. Wollen wir hoffen, daß die ST-Version dieses Backupprogramms hält, was die PC-Anleitung verspricht.

VIEL VERSPROCHEN

Was hat sich noch getan?... Da hätten wir zum Beispiel die Möglichkeit, bis zu acht Partitionen auf der HDplus zu installieren (bei den Modellen ab 80MB sogar bis zu deren sechzehn). Dann ist das Desktop des ST allerdings so schrecklich voll mit Diskicons (acht Icons für die Partitionen, zwei für Laufwerk A und B, eins für die RAMDisk, dazu noch diverse Papierkörbe), daß man kaum noch Platz für die Verzeichnisfenster findet. Eine Verbesserung, die unter die Kategorie KDV ("Kannste drauf verzichten") fällt. Man stelle sich nur das Chaos bei 16 Partitionen vor...

Desweiteren verspricht das Handbuch zur Platte allerlei Wunderdinge, wie etwa einen **Auto-Parker** und ein optionales **Cache-Memory.** Ein **Auto-Parker** ist ein Programm, das nach einer bestimmten (bei der HDplus einstellbaren) Zeitdauer den Schreib-/Lesekopf der Platte auf eine unbenutzte Spur fährt. Dort ist er einigermaßen ungefährlich für die Platte, falls ihr der Himmel auf den Deckel fällt oder so etwas.

Im **Cache-Memory** puffert die Platte einmal gelesene Sektoren, so daß die Suchzeiten beim erneuten Zugriff entfallen. Dafür muß man natürlich Hauptspeicher opfern.

VIEL GEHALTEN?

Zur Installation liefert Vortex auf der bereits vorformatierten Platte eine Utilitysammlung und ein Treiberprogramm. Zuerst formatiert man seine Platte physikalisch; dabei kann man noch einen Interleavefaktor angeben, der den Sektorversatz bestimmt. Laut Vortex ist hier eine Vier optimal; nach meinen eigenen Messungen kann man darüber streiten, aber dazu gleich.

Nach dem Formatieren wird partitioniert (ächz, noch ein Fremdwort), die Festplatte also in logische Laufwerke unterteilt. Dabei hat man auch eine recht große Auswahl unter Voreinstellungen. Zusätzlich bietet das Installationsprogramm eine CLEAR-Option, mit der man eine Partition logisch formatieren kann, was wesentlich fixer ist als das "richtige" Formatieren – das kann bei der HDplus 10 Minuten dauern!!

Die HDplus verwaltet (wie schon die alten Vortex-Platten) ihre defekten Sektoren selbst, das heißt, der integrierte Controller kümmert sich darum, daß Ersatzsektoren (oder gar ganze Spuren) zur Verfügung gestellt werden. TOS muß sich nicht mehr darum kümmern (indem etwa in der FAT bestimmte Cluster als defekt gekennzeichnet werden). Damit solche Sektoren erkannt werden, kann man die Option BAD (nein, nein, nein, nicht inspiriert von Michael "Kieks" Jackson) auf die Platte loslassen, was auch wieder fünf Minuten kostbarer Testerzeit kostet.

Und dann wäre da noch die BOOT-Option, mit der man erstens die Festplatte (von beliebigen Partitionen) bootbar machen, zweitens die **Auto-Park**-und die **Cache**-Option aktivieren kann. Die Größe des Cache-Speichers (der nach **first-in-first-out**-Prinzip verwaltet wird) kann man wiederum frei einstellen. Ein Glück, daß die HDplus diese **Auto-Park**-Option hat, denn das eigentlich im Handbuch versprochene Programm, das die Platte parken soll, war nirgendwo zu finden; so konnte ich vor dem Transport doch noch mein Gewissen beruhigen.

Tja, und nun alle Mann an die Stoppuhren.

TIME IS FLEETING...

Für meinen kleinen Benchmark ist ein Progrämmchen entstanden, das folgende Tests durchführt:

A) Hundertmaliges Lesen von 50 aufeinanderfolgenden Sektoren (per RWABS auf einen Schlag); dadurch wird das Laden von Programmen, die ja meist in aufeinanderfolgenden Sektoren liegen, recht gut nachgebildet.

B) Wie A, nur einzelne RWABS-Aufrufe; dies modelliert das Verhalten der Platte, wenn ein Anwenderprogramm beim Lesen von Sektoren ein schlechtes Timing hat. Ebenso kann man hier den Einfluß des Interleavings besonders gut beobachten.

C) Anlegen von 50 Dateien im Wurzelverzeichnis, nach jedem Anlegen Lesen eines Sektors von einer anderen Partition (Umpositionierung des Lesekopfes). Das bildet den Vorgang nach, wenn man kurze Dateien auf eine bereits leicht gefüllte Partition schreibt. Bei diesem Test erfaßt man ganz gut, wie stark die Zugriffszeiten auf die Geschwindigkeit der Festplatte einwirken.

D) Löschen der erzeugten 50 Dateien. Eigentlich ist dieser Teil nur im Programm, damit nach dem Ablauf die Partition wieder schön sauber aussieht. Aber da auch das ein Festplattenzugriff ist, wurde er natürlich mitgemessen.

Die Platte wurde probeweise mit verschiedenen Interleavefaktoren formatiert; der Test lief einmal mit aktiviertem Cache-Speicher (256 Sektoren) und dann ohne (Tabelle 1).

Vor dem Schrägstrich stehen die Zeiten (in Sekunden) ohne CacheSpeicher, danach diejenigen mit Cache-Speicher. Die Interleavefaktoren (ab jetzt wie in der Tabelle kurz IF) über 10 wurden nur teilweise ausgemessen, sie ergeben einfach zu schlechte Ergebnisse (ähnliches gilt für IF 2). Beim Faktor 1 meldete sich das Formatierprogramm mit einer Fehlermeldung, es könne den Konfigurationssektor nicht mehr lesen. IF 5 und 9 liegen im restlichen Trend und sind deswegen

nicht extra aufgeführt, um die Tabelle nicht zu überfrachten.

Auffällig: Test B hat optimale Zeiten bei IF 8, Test A dagegen bei Interleave 3. Die Zeiten für Test C liegen sehr eng beieinander, bei Test D ist wieder der IF 3 optimal. Wie man seine Platte formatiert, hat also oft unerwartete Auswirkungen auf die Geschwindigkeit bei verschiedenen Anwendungen. Ich persönlich würde die Vortex-Platte mit IF 3 formatieren.

Bei den verschiedenen Interleavefaktoren habe ich auch den schon fast klassischen **WordPlus-Test gemacht,** das heißt, WordPlus samt allen dazu nötigen Paraphernalien von Diskette auf eine Partition kopiert, dann gestartet und schließlich in einen anderen Ordner geschrieben. Hier hatte der IF 3 wieder klar die Nase vorn. Ein Kuriosum: Bei IF 6 war das Kopieren in einen anderen Ordner mit aktiviertem Cache-Speicher langsamer als ohne! Überhaupt spart der Cache-Speicher im alltäglichen Einsatz wohl selten mehr als zehn, maximal 20 Prozent Zeit. Für mich kommt so ein Cache-Speicher nicht in Frage; lieber installiere ich mir eine RAM-Disk und lasse die zeitkritischen Arbeiten darin ablaufen.

Übrigens war die HDplus beim WordPlus-Benchmark etwas langsamer als die SH205 von ATARI, ebenso bei den Tests A und B (siehe oben), bei den Tests C und D allerdings schneller.

IM SCHWEISSE MEINES ANGESICHTES

... saß ich einige Tage vor der HDplus und versuchte herauszufinden, wie sie sich für die Programmier- und Anwendungsarbeit eignet.

Vortex verspricht im Handbuch einen leisen Lüfter, denn "auch das gehört zu einem überzeugenden System" (O-Ton Vortex). Ganz meiner Meinung, nur konnte ich den leisen Lüfter in der HDplus nicht entdecken, weil der große laute darin alle anderen Geräusche überdeckte: Beim Einschalten kommt Honda-Feeling auf, danach legt sich der satte Sound etwas, so daß er gerade erträglich wird (zumindest schon eher als der Windkanal in der SH205). Dieses Lüfter-Dilemma werde ich nie verstehen, bei den PCs geht's doch auch.

Weiterhin soll die Zusammenarbeit mit der SH205 von ATARI besonders gut funktionieren; immerhin sind ja schon die AdressDIPs herausgeführt, so daß man nicht das Gehäuse aufschrauben und die Garantie verletzen muß, um die Platte auf einer anderen Adresse ansprechen zu können. Wenn ich das Handbuch richtig verstanden habe, kann man mit den sechs DIP-Schaltern zwischen jeweils 8 Subadressen der Adapteradressen 0 und 1 umschalten, das heißt, da meine SH205 auf Adapter 0 eingestellt ist, muß man zum gleichzeitigen Betrieb die HDplus auf Adapter 1 umschalten.
So ganz einwandfrei funktionierte die Zusammenarbeit allerdings bei mir nicht immer. Mal formatierte mir das Installationsprogramm der HDplus statt der HDplus meine SH205 (von der ich in weiser Vorahnung einen Backup gezogen hatte), mal mochten sich die beiden Treiber nicht sonderlich, mal wollte die HDplus erst gar nicht auf Ansprache reagieren.

Das Handbuch läßt einen in diesem Punkt ziemlich im Stich; wahrscheinlich gibt es da draußen in der großen weiten Welt noch so ein paar schußlige Anwender wie mich, so daß der gleichzeitige Gebrauch von SH205 (SH204) und HDplus nicht ganz ungefährlich ist.

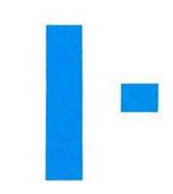
Überhaupt sollte man sich bei Vortex gründliche Gedanken über das neue Handbuch machen. Was Dokumentation betrifft, sollte man doch bitte keinen Wert auf Kompatibilität mit ATARI legen. Das alte Handbuch war jedenfalls wesentlich umfangreicher und informativer (auch und vor allem was die Interna der Platte angeht). Aber da auf dem Titelblatt noch das Wörtchen "vorläufig" prangt, besteht noch Hoffnung.

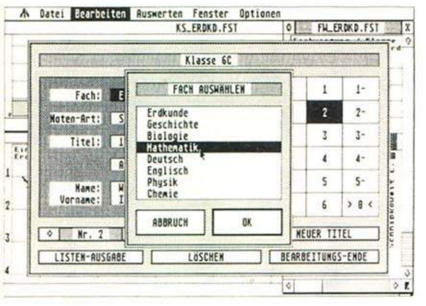
Gefallen hat mir die Intelligenz der Treibersoftware. Der **Auto-Parker** (sowas bräuchte man mal im Stadtverkehr) versah seinen Dienst pünktlich und zuverlässig, indem er den Lesekopf (in meinem Fall) alle 30 Sekunden beiseite schaffte. Leider fährt die Platte den Kopf aber nicht in Parkposition, wenn man sie ausschaltet (manche Platten nutzen den Ausschaltimpuls, um das kurz vor dem Exitus noch zu erledigen). So muß man also nach dem letzten Zugriff immer noch eine Weile warten, bis die Platte durch rhythmisches Geblinke anzeigt, daß sie einen Parkplatz gefunden hat. Auf das Cache-Memory dagegen kann ich, wie schon gesagt, leichten Herzens verzichten.
Der Controller in der HDplus ist ja recht intelligent, verwaltet er doch seine Fehlsektoren selbst. Zudem versteht er auch erheblich mehr Kommandos als andere Festplattencontroller, was aber im Normalbetrieb ziemlich nebensächlich ist. Trotzdem muß man enttäuscht sein, daß man diesen Komfort offensichtlich ab und an mit kleinen Geschwindigkeitseinbußen bezahlen muß (siehe Benchmarks).

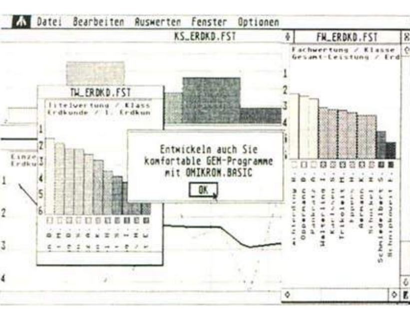
Fazit: Wenn das Handbuch erst einmal so solide ist wie die Platte selbst, wenn zudem die versprochenen BACKUP- und PARK-Programme auch wirklich mitgeliefert werden, ist die HDplus auch zu einem Preis von 1298,– DM ein erwähnenswerter Konkurrent für die SH205.

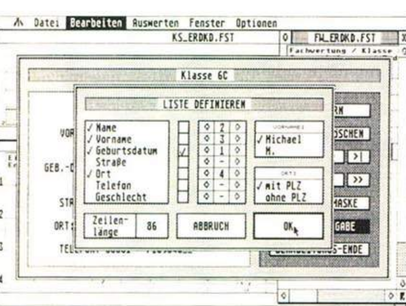
(Claus Brod)

Vortex Computersysteme GmbH
Falterstr. 51-53
7101 Flein
Tel.: 07131/52061









GEM-Programme ohne Wenn und Aber!

Mit OMIKRON.BASIC können Sie ohne Pokes oder sonstige Verrenkungen Applikationen entwickeln, die vollständig in GEM eingebunden sind. Alle erforderlichen Prozeduren, deren Namen den C-Konventionen entsprechen, stellt Ihnen unsere GEM-Library zur Verfügung.

ARITHMETIK: Rechengenauigkeit bis 19 Stellen bei allen Funktionen ● Rechenbereich bis 5.11 E±4931

GEM: Komplette GEM-Library ● Sämtliche AES- und VDI-Funktionen direkt mit Namen verfügbar ● Eigene BITBLIT-Routine

EXTRAS: Masken-INPUT ● SORT-Befehl sortiert beliebige Felder, auch mit Umlauten ● Matrizenbefehle

STRUKTUR: Prozeduren und mehrzeilige Funktionen mit Übergabe- und Rückgabe-Parametern und lokalen Variablen ● REPEAT... UNTIL, WHILE... WEND, mehrzeiliges IF... THEN... ELSE... ENDIF ● Labels bei GOTO, GOSUB, ON GOTO und ON GOSUB

EDITOR: Mit oder ohne Zeilennummern (umschaltbar) ● Drei Schriftgrößen bis 57 x 128 Zeichen ● Frei definierbare Teach-In-Funktionstasten

GESCHWINDIGKEIT: FIT-Code (FIT = Fast Interpreting Technique) ● Wir kennen keine schnelleren 68000-Fließkomma-Routinen ● Volle Integer-Arithmetik ● Eigene Disk-Routinen für beschleunigten Dateizugriff

KOMPATIBILITÄT: Für alle ATARI ST (auch MEGA ST) ● 99 % MBASIC-kompatibel ● Editor findet Inkompatibilitäten, dadurch einfachste Anpassung

● Diskettenversion mit Demodiskette und Handbuch **DM 179,–**

● Modulversion mit Demodiskette und Handbuch **DM 229,–**

COMPILER

Noch mehr Tempo erreichen Ihre mit OMIKRON.BASIC entwickelten Applikationen durch unseren OMIKRON.BASIC-COMPILER.

Er erlaubt es Ihnen, bequem in Basic interpretativ zu programmieren, um nach der Compilierung Geschwindigkeiten zu erreichen, die bisher C-Compilern vorbehalten waren. Der Aufbruch in neue Dimensionen der Basic-Programmierung hat begonnen.

● Diskette mit Anleitung **DM 179,–**





... denn das Beste ist für Ihren ATARI ST gerade gut genug!

OMIKRON.SOFTWARE

Erlachstraße 15 · D-7534 Birkenfeld · ☎ (0 70 82) 53 86

Frankreich: OMIKRON · France, 11 Rue Dérodé, F-51100 Reims
Luxemburg: ELECOMP, 11 Avenue de la gare, L-4131 Esch-Alzette
Niederlande: Terminal Software Publicaties, Postbus 111, NL-5110 Baarle-Nassau
Österreich: Ueberreuter Media, Laudongasse 29, A-1082 Wien
Schweiz: Thali AG, Industriestraße 6, CH-6285 Hitzkirch

Der Versicherungsprofi

Ein Erfahrungsbericht über das neue Versicherungspaket V_MANAGER prosy

Nach langem Warten ist nun endlich ein Programm für die Verwaltung von Versicherungsagenturen auch für die ATARI ST Computer auf den Markt gekommen. Nach Aussage des Herstellers wurde das Programm in Zusammenarbeit mit mehreren Agenturen entwickelt und soll deren Bedarf in allen Punkten der Verwaltung und Akquisition abdecken. Ob das Programm diesen hohen Ansprüchen gerecht werden kann, wird dieser Erfahrungsbericht zeigen.

Als EDV-Neuling war für mich die Qualität des Handbuches von besonderer Bedeutung, da ich im Umgang mit dem Atari ST noch nicht sehr vertraut war. Das ca. 120 Seiten umfassende Handbuch führt den Leser Schritt für Schritt durch das Programm. Schon nach dem Durcharbeiten der ersten Seiten, konnte ich das Programm in Betrieb nehmen und mich nach und nach mit den einzelnen Menüpunkten vertraut machen. Dabei war es gut möglich, das gesamte Programm kennenzulernen, da das Handbuch in der gleichen Reihenfolge angeordnet ist, wie die Vorgehensweise bei der Erstellung eines Kundenstammes.

V_MANAGER prosy ist durch ein sogenanntes Hardwaremodul kopiergeschützt, welches vor dem Einschalten des Rechners in den Modulport gesteckt werden muß. Vergißt man dieses, wird das Programm unmittelbar nach dem Start wieder verlassen. Das Modul kann auch beim Arbeiten mit anderer Software eingesteckt bleiben, dadurch entfällt ein lästiges Einstecken des Schutzmodules vor dem Arbeiten mit V_MANAGER.

Ein wenig extravagant präsentiert sich die Benutzeroberfläche des Programmes. Man hat sich sogar die Mühe gemacht und die Hauptdatenblätter, sowie die Vertragsordner mit einem Grafikprogramm erstellt, um dem Anwender ein möglichst realitätsnahes Arbeitsumfeld bieten zu können. Ich meine, daß dieses hervorragend gelungen ist.

Arbeiten mit dem V_MANAGER

Gestartet wird das Programm aus der Directory heraus wie üblich durch Anklicken des Programmfiles „V_M.PRG“. Eine Box für die aktuelle Datumseingabe erscheint. Das Datum wird nach der Eingabe in einer Statusbox geführt, die außerdem noch einige andere für den Anwender wichtige Informationen verwaltet, wie z. B. die Anzahl der Kundendatensätze und der eingetragenen Verträge. Alle Kunden- und Versicherungsdaten werden in einer Datei gespeichert, die beim erstmaligen Benutzen von V_MANAGER angelegt werden muß, und dann immer zur Verfügung steht. Es können sogar mehrere Dateien angelegt und bearbeitet werden. Für die Kundendaten stehen zwei Datenblätter zur Verfü-

Jetzt bringt der ATARI ST Ihr Büro in Form

ST-Kontor
Kaufmännische Anwendungen für alle ATARI ST-Computer
ST-Kontor
TOS-Manager
Kundenverwaltung
Lager + Fakturierung
Finanzbuchhaltung
Lohn + Gehalt
SYBEX
Finanzbuchhaltung
Software + Handbuch

Wußten Sie eigentlich, wieviel Arbeit Ihnen Ihr ST bei den täglichen Büro-Aufgaben abnehmen kann? Immer vorausgesetzt, daß Sie über eine Software verfügen, die alle nötigen Funktionen besitzt, komfortabel und schnell ist; wie die von der GFE R. Becker KG entwickelten Programme der Reihe SYBEX ST-Kontor (jeweils mit Trainingsbuch). Ihr ATARI ST (260–1040, MegaST 2–4) sollte einen Speicher von 1 MB RAM (für TOS-Manager und Kundenverwaltung reichen auch 512 KB) und das Betriebssystem TOS im ROM haben. Die mächtige FiBu benötigt ein double-sided Laufwerk mit 1 MB, optimal wäre eine Festplatte.

Natürlich finden Sie bei SYBEX noch mehr Software und Bücher zu Ihrem Rechner. Fordern Sie einfach unseren ATARI ST-Prospekt oder unser Gesamtverzeichnis an.

So hilft Ihnen der ST-Kontor TOS-Manager:

Er ist die neue Verwaltungszentrale Ihres ATARI-Rechners. Von hier aus können Sie alle übrigen ST-Kontor-Programme ohne langes Suchen aufrufen. Viele Accessories erleichtern Ihnen die Arbeit: RAMDisk, Taschenrechner, Terminkalender, ASCII-Tabelle, Notizblock usw. Die Diskettenverwaltung ersetzt die GEM-Benutzeroberfläche und bietet ihnen viele zusätzliche Möglichkeiten, wie: Formatieren und Kopieren von Disketten, Bearbeiten von Ordnern und Dateien, Suchen von Dateien, Restaurieren gelöschter Dateien, Druckereinstellung. Ein Programm, mit dem Sie Ihre tägliche Arbeit einfacher und effizienter gestalten.

Best.-Nr. **3428, DM 98,–*/sFr. 98,–/S 872,–**

Und das kann die ST-Kontor Kundenverwaltung:

Als Basismodul der ST-Kontor-Reihe ist die Kundenverwaltung für die Verwaltung Ihrer Adressen-Bestände zuständig; nur Ihre Festplatte bzw. das RAM können dabei Grenzen setzen. Die wichtigsten Features im Überblick:

- Superschnelles Suchen (Suchen nach Von-Bis-Werten in max. 0,5 Sekunden)
- Optimierte Datenspeicherung
- Mehrere Datenfelder pro Adresse (Stammdaten, Zusatzdaten, Notizen u. a.)
- Umfangreiche Selektionsmöglichkeiten nach beliebig vielen Kriterien
- Ausgabe auf Datei/Monitor/Drucker als Liste, auf Etikett, als Serienbrief

Best.-Nr. **3429, DM 149,–*/sFr. 149,–/S 1326,–**

Ein Bündel arbeitssparender Features enthält die ST-Kontor Lagerverwaltung & Fakturierung:

Mit diesem starken Kombi-Programm bringen Sie Ihr Sortiment auf Vordermann: Stücklisten, Etiketten, Preislisten, Kataloge und umfangreiche Angebote mit Textbausteinen sind jetzt kein Problem mehr für Sie. Ebenso selbstverständlich sind verschiedene Lagerbestands-Bewertungsmethoden, Übersicht über den aktuellen Lagerbestand mit Bestellvorschlägen, Fakturierungsübersicht mit Offenen Posten, Teilen und Zusammenlegen von Fakturierungen – und die Übernahme in die Finanzbuchhaltung. Nicht zu vergessen die Bildschirmkasse mit dem aktuellen Kassenstand, Anzeige des Wechselgeldes und Ausdruck eines Kaufbelegs. Und betriebswirtschaftliche Funktionen zur Betriebsoptimierung werden direkt mitgeliefert.

In Vorbereitung

Best.-Nr. **3430, DM 398,–*/sFr. 398,–/S 3542,–**

Den aktuellen EG-Richtlinien entspricht die ST-Kontor Finanzbuchhaltung:

Die mehrfirmen- und mandantenfähige FiBu verdaut große Buchungsmengen mit einer erstaunlichen Geschwindigkeit. Unter anderem können Sie von ihr erwarten:

- Kontenrahmen (DATEV-Kontenrahmen SKR 04) mit 6000 Konten
- Verschiedene Bilanzierungs- und Abschlußmöglichkeiten mit einfacher Kostenstellen-Rechnung, Anlagenspiegel, Abschreibungs-Verzeichnis und betriebswirtschaftlicher Auswertung
- Verbuchen von Sammelbelegen mit maximal 10 Soll- und 10 Habenkonten sowie 4 Mehrwertsteuer-Sätzen pro Buchung
- Abwicklung des Zahlungsverkehrs mit Offenen Posten und Mahnwesen
- Übernahmedatei für Fakturierung/Lohn & Gehalt mit Korrekturmöglichkeiten

Best.-Nr. **3431, DM 498,–*/sFr. 498,–/S 4432,–**

Das erledigt ST-Kontor Lohn & Gehalt für Sie:

- Verwaltung der Mitarbeiterdaten samt Ausfüllen aller Versicherungs- und Finanzamtsformulare
- Alle Lohn- und Gehaltskonten auf einen Blick
- Verwaltung von Zeitkonten einschließlich Krankheits- und Urlaubstagen
- Abrechnung von Vermögensbildung, Direktversicherung und Lohnpfändung
- Komplette Erstellung der Lohnsteuer-Jahresdaten ohne Rechenaufwand
- Abwickeln des Zahlungsverkehrs mit Mitarbeitern, Ämtern und Versicherungen inklusive Berechnung der Zahlungsbeträge und Vorbereitung der Buchung. Die Daten können von der ST-Kontor FiBu übernommen und weiter bearbeitet werden.

In Vorbereitung

Best.-Nr. **3432, DM 198,–*/sFr. 198,–/S 1760,–**

*Unverbindliche Preisempfehlung

Übrigens: SYBEX sucht ständig gute Buch- und Software-Autoren. Interessiert? Dann kontakten Sie bitte Ralf Lieder, Tel. 0211/618 02 20.



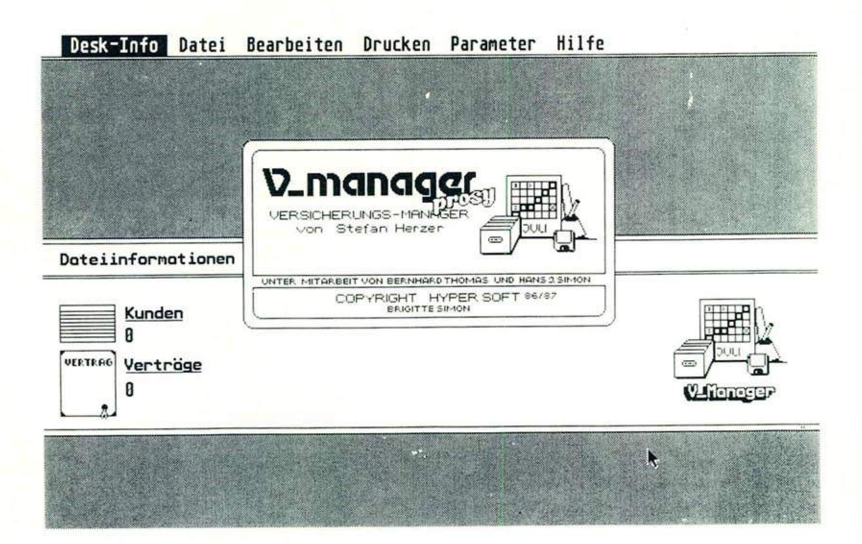


Bild 1: So präsentiert sich der V_MANAGER

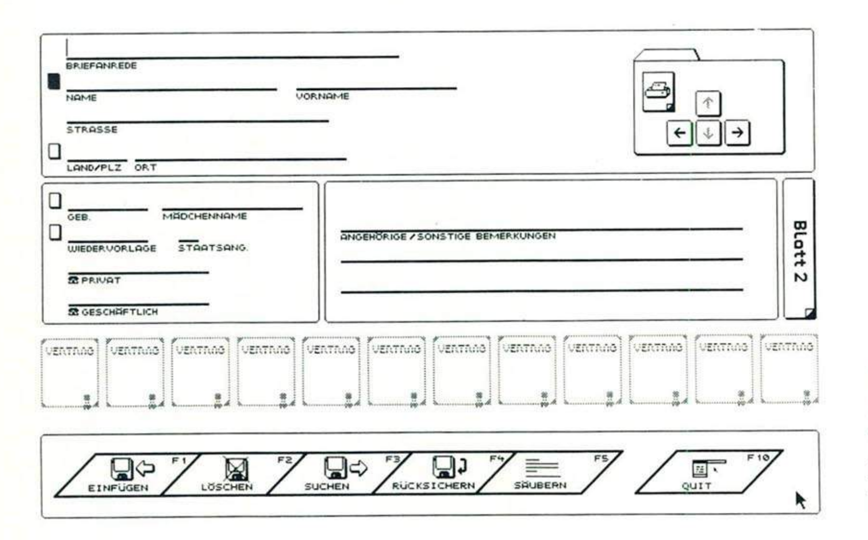


Bild 2: Die Unterteilung des Hauptdatenblattes

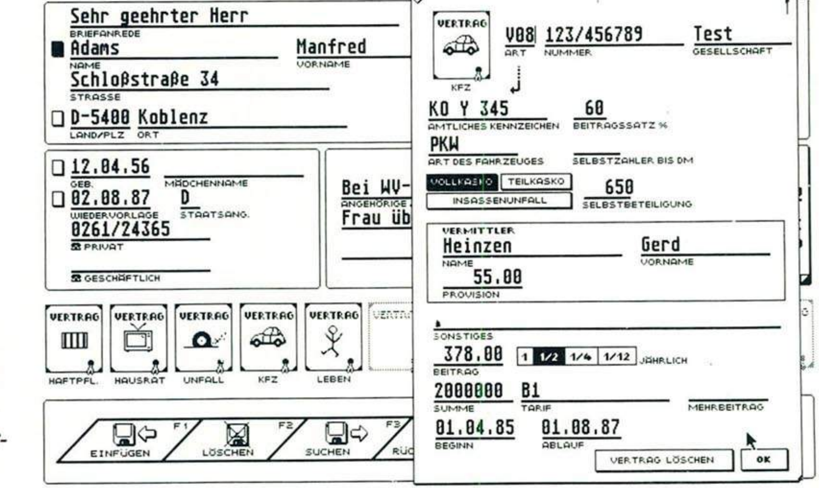


Bild 3: Ein Versicherungsordner am Beispiel einer Kfz-Versicherung

gung, die sauber getrennt für die allgemeinen und die spezifischen Personaldaten geführt werden. Die Versicherungsdaten werden übersichtlich im Ordnerkonzept angelegt und stehen auf Tastendruck, nach Ihren Arten (Haftpflicht, Hausrat etc.) geordnet zur Verfügung. Die Ordner sind in drei Bereiche eingeteilt. Zu den allgemeinen Vertragsdatenfeldern sind explizit in jedem Ordner die spezifischen Vertragsdaten-, sowie Vermittlerdatenfelder implementiert worden. Über die Funktionstasten lassen sich alle Vertragsarten abrufen und bearbeiten. Jeder schon bearbeitete Vertragsordner wird auf dem Hauptdatenblatt symbolisch dargestellt, und mit einem entsprechenden Untertitel versehen. Das heißt, Sie sehen schon auf den ersten Blick, welche Versicherungen ein Kunde hat. Gerade in diesem Punkt unterscheidet sich das Programm von seinen Konkurrenten der alten Garde, eine so bestechende Übersicht ist mit einem textorientierten Programm kaum möglich. Eine schnelle Auskunftsbereitschaft wird somit gewährleistet. Alle Ordner und Datenfelder sind aus der Praxis entnommen worden, und ich konnte aus dem großen Angebotsrepertoir an Datenfelder und Buttons keines finden, das nicht seinen Zweck erfüllt hätte.

Dennoch habe ich den Eintrag von mitversicherten Personen oder die Möglichkeit eines Änderungsdatums vermißt. Aber auch das, so bestätigte mir die Firma HYPER SOFT, sei in der jetzt lieferbaren Version integriert. Nachdem ich meinen Kundenbestand in V_MANAGER eingearbeitet hatte, konnte ich mich erst so richtig von der Effektivität und dem Nutzen dieses Agenturpaketes überzeugen. Auf „Maustastendruck" stehen einem aktuelle Statistiken und Bestandszahlen zur Verfügung, Provisionsrechnungen und Terminpläne werden individuell gedruckt. Vielseitiger Schriftverkehr ist problemlos möglich. Ob Serienbriefe oder Einzelbriefe, Angebote oder sonstige Formulare, alle können vom Programm aus gefertigt und gedruckt werden. Auf der Originaldiskette befinden sich eine Vielzahl von Standardformularen, wie z. B. eine Datensatzliste mit allen wichtigen Personal- und Vertragsdaten, eine Beitragsliste, eine Adreßliste, eine Telefonliste u.v.m., die sofort benutzt werden können. Desweiteren besteht die Möglichkeit alle nur erdenklichen Formulare oder Listen mit einem Listengenerator selbst zu fertigen. Die Formulare und Listen werden in einem Textprogramm mit dem Listengenerator erstellt und können dann direkt weiter verarbeitet werden. Ein Verlassen des Hauptprogrammes ist nicht mehr notwendig.



JETZT SPIELEN WIR ALLE TRÜMPFE AUS.
GEMAUERT WIRD NICHT!
EPROM-Programmiergerät:
programmiert EPROMs der 27er Serie von 2764 bis 27256 sowie die entsprechenden CMOS-Typen. Die Einstellung sämtlicher Spannungen erfolgt softwaremäßig. Komplett mit GEM-Software.
DM 249,-
EPROM-Karte 128 KB
128 KB Speicherkapazität. Per Schalter ein- und ausschaltbar. Industriell gefertigt. Mit optischer Zustandsanzeige. Ausführliche deutsche Anleitung.
DM 39,-
Sound-Sampler
10 Bit (!), Musik- und Sprachausgabe bis zu 15 Minuten (bei 1 MB RAM), inkl. 3 Software-Paketen, mit allen Anschlußkabeln sowie seperatem Netzteil, Ausführliches deutsches Handbuch
DM 298,-
jetzt neues Design
NEUE PREISE
Die Winteroffensive hat begonnen!
Neu: Monitor-Umschaltung
Monochrom/Farbe
Endlich: Kein lästiges Umstecken mehr. Stecker leiern nicht mehr so schnell aus.
jetzt nur DM 39,90
Uhr-Modul
Ausgabe von Datum und Uhrzeit jederzeit möglich. Daten bleiben nach dem Ausschalten des Rechners durch Akkupufferung erhalten. Keine Löt- oder Schraubarbeiten erforderlich. Ausführliches deutsches Handbuch.
DM 69,-
JOKER
Der Super - Trumpf:
eine scharfe Sache für Ihren Fernseher
PAL - INTERFACE III
- ein Feuerwerk der Farben, Darstellung aller 512 möglichen Farben an jedem Farbfernseher.
- jetzt auch Mischfarbendarstellung uneingeschränkt möglich.
- seperate Ausgänge für:
- Antennenbuchse des Fernsehers
- Audioeingang der Stereoanlage
- Videoeingang des Monochrom-monitors/Videorecorders
- lauffähig an allen ST-Versionen
- Lieferung komplett anschlußfertig mit allen Kabeln sowie seperatem Netzteil
- ausführliche deutsche Anleitung.
nur DM 198,-
Weiteres Zubehör:
Zusatzgehäuse aus Metall DM 148,00
Monitor-Verlängerungs-Kabel DM 55,00
Floppy-Verlängerungs-Kabel DM 49,00
Druckerkabel ATARI/Centr. DM 39,00
Monochrom-Monitor-Adapter DM 55,00
Diskettenbox für 50 Disketten DM 29,90
Staubschutzhaube, tranparent, rauchglasfarben DM 29,90
Experimentierplatine für den Expansionsport DM 19,90
Disketten 3,5'', 100% Error free, single sided, pro Stück DM 3,40
EPROM 2764 (8 KB) DM 6,90
EPROM 27128 (16 KB) DM 8,90
EPROM 27256 (32 KB) DM 9,90
Joystick 'Competition Pro' mit Mikroschaltern DM 49,00
EPROM-Löschgerät für bis zu 12 EPROMs gleichzeitig DM 115,00
Floppy-Stecker DM 9,90
Monitor-Stecker DM 8,50
und vieles mehr...
Digitalis
Vier Geräte in einem: Oszilloskop, Oszillograph, 10-BIT-A/D-Wandler und Speicheroszilloskop. 86 kHz Abtastfrequenz. GEM-Software, ausführliches deutsches Handbuch.
DM 298,-
Händleranfragen erwünscht.
Achtung: Neue Adresse!
Computertechnik
Z. Zaporowski
Dreieckstr. 2b
5800 Hagen 1
Tel. 02331 / 86555
Vertrieb in der Schweiz
MFS. Sägesser
CH-31 85 Schmitten
Tel. 037 / 36 20 60
Fordern Sie unseren Gratiskatalog an!

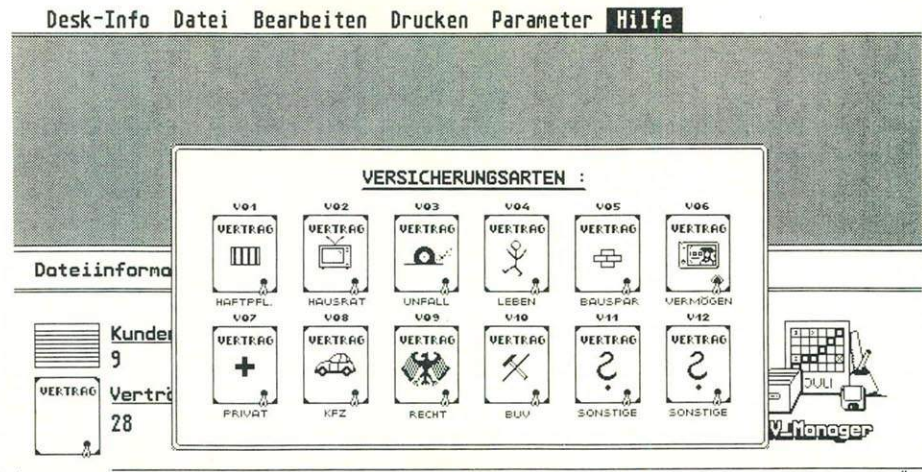


Bild 4: Die Versicherungsarten

Ausgefeilte Selektion

Alle Listen und Formulare können an jeden einzelnen Kunden direkt, an alle Kunden aus dem Bestand, oder aber selektiert an bestimmte Kundenkreise gerichtet werden. Letzteres wird über eine sehr Umfangreiche Selektion erreicht. Es stehen Ihnen über 25 verschiedene Kriterien zur Verfügung, die mit „und", „oder" sowie „nicht" verknüpft werden können. So ist es z. B. möglich einen Kundenkreis anzusprechen, der in einem bestimmten Gebiet wohnt, ein bestimmtes Alter und Monatseinkommen hat und noch keine Lebensversicherung besitzt. Sogar nach Fremdversicherungen kann selektiert werden. Die Selektion ist ein Werkzeug, das man nach einiger Zeit nicht mehr missen möchte.

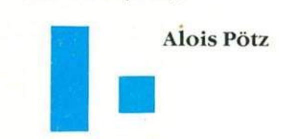
Um das Programm professionell einzusetzen, schreibt der Hersteller die Benutzung einer 20 MB-Festplatte vor, und dies mit gutem Grund, denn der Geschwindigkeitsunterschied zu einem Laufwerk ist doch sehr erheblich. Auch nach 600 eingegebenen Kundendatensätze konnte ich keine merklichen Verzögerungen bei den Dateioperationen feststellen.

Um letztendlich alles zu erwähnen, was dieses Programm zu leisten vermag, oder wie geschickt man verschiedene Funktionen bei der täglichen Arbeit einsetzen kann, reicht dieser kurze Bericht natürlich nicht aus. Ich kann allen Interessenten nur empfehlen, sich beim ATARI ST Fachhandel um mehr Informationen oder eine Vorführung des Systems zu bemühen. Das Preis/Leistungsverhältnis ist noch besonders hervorzuheben, mir ist jedenfalls derzeit kein vergleichbares System bekannt.

Preis: 1498,– DM incl. MwSt.

Hersteller: HYPER SOFT
Hauptstraße 44
5441 Auderath
Tel. 02676/1863

Alois Pötz



Das Acht-Damen-Problem

Die Benchmark-Tests für KI-Sprachen in der Juni-Ausgabe dieser Zeitschrift gaben mir den Anlaß, mich intensiver mit dem Acht-Damen Problem zu beschäftigen.

Ich möchte hiermit einen Vergleich der konventionellen optimierten Programmierung mit den KI-Sprachen ermöglichen. Dieser Vergleich soll keinesfalls den Nutzen dieser Sprachen bestreiten, es soll vielmehr gezeigt werden, wie groß die Leistung herkömmlicher Programmiersprachen für klar umrissene Probleme ist. Das hier vorgestellte Programm ist etwa um den Faktor 1000 (in Worten eintausend) schneller als die im Juni vorgestellten Lösungen.

Desweiteren soll an einem einfachen Beispiel gezeigt werden, wie die Umsetzung eines Problems in ein das Problem lösendes Programm erfolgt. Es wird das Prinzip der Rekursion erläutert und ein kurzer Einblick in Bitoperationen bei der Programmiersprache 'C' gegeben.

Problemdefinition

Das Acht-Damen-Problem beschäftigt sich, wie der Name schon sagt, mit acht Damen (Schachfiguren). Diese sollen so auf einem Schachbrett positioniert werden, daß sie sich gegenseitig nicht schlagen können. Die Bewegungsmöglichkeiten sind die einer Dame im Schachspiel (senkrecht, waagerecht und diagonal).

Lösungsansatz

Unser Programm könnte jetzt natürlich einfach alle acht Damen über das gesamte Schachbrett laufen lassen und abprüfen, ob die Damen erstens nicht auf identischen Positionen stehen und zweitens, daß sie sich gegenseitig nicht schlagen können.

Dieses Programm besteht aus acht ineinander geschachtelten Schleifen, die je einer Dame zugeordnet sind und diese über alle 64 Felder laufen lassen. Der damit verbundene Zeitaufwand liegt bei unserem Rechner bei etwa 7000 Jahren (optimistisch); ein wohl offensichtlich indiskutabler Zeitraum. Es obliegt also das Problem der Optimierung.

Optimierung

Wenn man sich an ein Schachbrett setzt und versucht, die Damen darauf zu verteilen, stellt man schnell fest, daß es sinnlos ist, mehr als eine Dame in einer Reihe aufzustellen.
Diese Erkenntnis nutzen wir, indem wir unser Schachbrett nicht etwa als Matrix mit acht mal acht Plätzen vereinbaren, sondern für jede Reihe nur ein Feldelement benutzen; der Wert, der in dem Feldelement steht, gibt uns hierbei die Position der Dame in der Reihe an (s. Abb. 1).

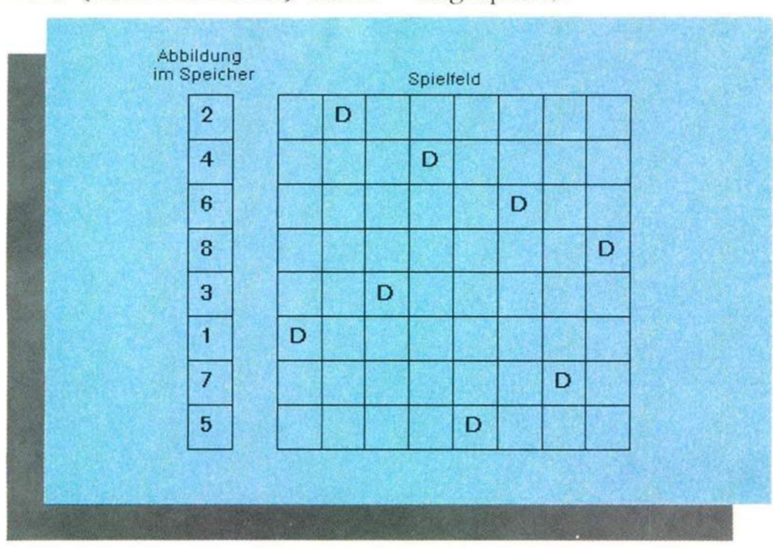


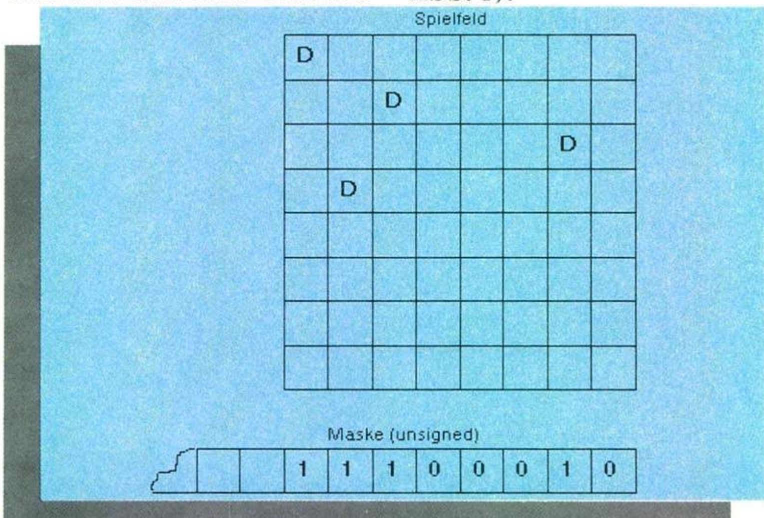
Abb. 1:
Die Damen bewegen sich jeweils nur in einer Reihe; für jede Reihe wird ein <unsigned> reserviert, unser Feld (acht Elemente) kann so das gesamte Spielfeld repräsentieren.
Die Werte in der Abbildung geben die Spalte an, in der sich die Dame befindet. Die Speicherrepräsentation geht aus Abb. 4 hervor.

Unsere Schleifen müssen nun nicht mehr über das gesamte Spielfeld laufen, sondern nur noch über die der Dame zugehörige Reihe (also nicht von eins bis 64, sondern von eins bis acht). Der Zeitaufwand verkürzt sich damit auf ca. 1600 Sekunden (wie oben wird von 10.000 Stellungsüberprüfungen pro Sekunde ausgegangen).

Eine weitere Möglichkeit der Optimierung ergibt sich, wenn man sich das Setzen der Damen in verschiedene Reihen auf dem Schachbrett vorstellt. Was für die Reihen gilt, gilt in gleicher Weise auch für die Spalten; es darf nur jeweils eine Dame gesetzt werden (die Überprüfung der Diagonalen folgt später).

Richten wir uns also eine Maske ein, die die bereits besetzten Spalten repräsentiert. Da ich als Implementationssprache 'C' gewählt habe ist das auf eine sehr einfache Weise möglich: Man nimmt ein <unsigned> und setzt das der Spalte entsprechende Bit (s. Abb. 2).

Abb.2:
Unsere Maske repräsentiert die durch Damen besetzten Spalten (siehe auch Abb. 4).



Unser Programm prüft nun ganz einfach, ob in der Spalte, in die wir unsere Dame setzen wollen, das entsprechende Bit gesetzt ist. Ist dies der Fall, so brauchen wir uns um die tieferen Reihen nicht mehr zu kümmern (es gibt ab dieser Reihe sowieso immer zwei Damen, die sich schlagen können).

Jetzt sind wir am einem Punkt angekommen, an dem es sinnvoll ist, die Möglichkeiten der rekursiven Programmierung (s. Abb. 3) zu nutzen.

Wir lassen unsere geschachtelten Schleifen das sein, was sie sind (schön und anschaulich), und begeben uns auf das Glatteis der Rekursion. Hierzu brauchen wir eine Funktion, deren Aufgabe es ist, eine Dame durch eine Reihe zu schieben (s. Abb. 4) und abzuprüfen, ob diese Dame von einer vorher plazierten geschlagen werden kann. Ist dies der Fall, so kann keine Kombination der Damen in den tieferliegenden Reihen eine Lösung ergeben. Kann die neu plazierte Dame jedoch nicht geschlagen wer-

den, ruft sich die Funktion rekursiv (bzw. selbst) auf und erzeugt damit Damenstellungen in den tieferen Reihen. Der Ebenenparameter (vgl. Abb. 3) bei diesem Aufruf ist natürlich die tiefere Reihe. In ihr wird der beschriebene Vorgang wiederholt.

Lising: rekursive Funktionsaufrufe einfügen

Abb. 3:
Rekursion bedeutet, daß eine Funktion sich selbst, evtl. über Zwischenaufrufe, aufruft. Das Prinzip ähnelt dem von geschachtelten Schleifen, hat jedoch den wesentlichen Vorteil, daß die Zahl der Schachtelungen nicht vorher festgelegt werden muß.
Die Anzahl der Schachtelungen wird durch die Rekursionstiefe bestimmt; außerdem werden in jeder Rekursionsebene die gleichen Operationen ausgeführt (nicht, wie bei geschachtelten Schleifen, nur in der innersten).
Damit die Funktion sich nicht unendlich oft selbst aufruft, muß sie überprüfen, wie oft sie bereits geschachtelt wurde; dies wird durch die Übergabe eines Parameters erreicht, der die aktuelle Rekursionstiefe (in gewünschter Form) enthält. Bei jedem Selbstaufruf der Funktion sollte dieser Parameter verändert werden, damit sich die tiefer geschachtelte Funktion über ihre Schachtelungsebene im klaren ist.

```
                    rekursive Funktionsaufrufe

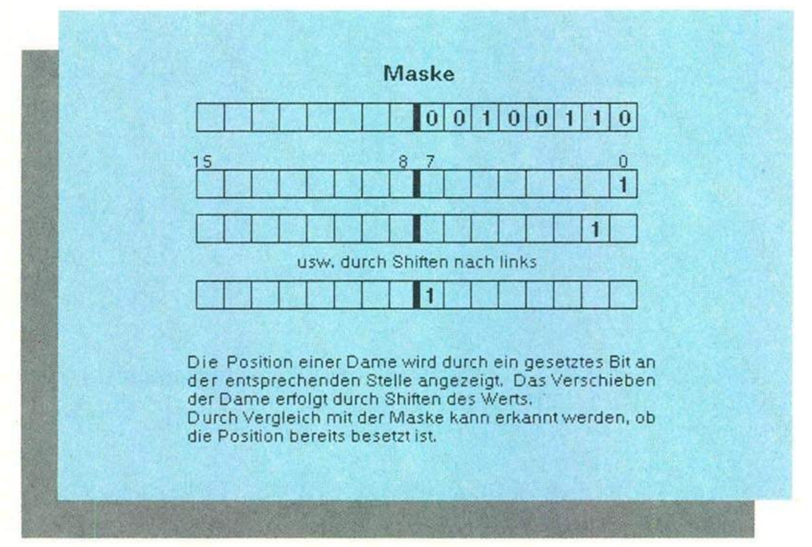
    Routine_A(Ebene) {
        IF (Ebene >= maximale Rekursionstiefe) Abbruch;
        ELSE Routine_A(Ebene+1);
    } /* einseitige Rekursion */


    Routine_1(Ebene) {
        IF (Ebene >= maximale Rekursionstiefe) Abbruch;
        ELSE Routine_2(Ebene);
    } /* wechselseitige Rekursion, Funktion 1 */

    Routine_2(Ebene) {
        Routine_1(Ebene+1);
    } /* wechselseitige Rekursion, Funktion 2 */
```

Die Rekursion wird abgebrochen, wenn sie ihre maximale Tiefe erreicht hat. Ein Abbruch bedeutet, daß die weitere Schachtelung beendet wird, als nächstes also die vorher aktive (aufrufende, höhere) Funktion aktiviert wird. Wird die oberste Rekursionsebene verlassen, so ist die Rekursion beendet. Mit jedem Funktionsaufruf innerhalb der Rekursion ist ein Ebenenwechsel verbunden. Rekursion ist das grundlegende Prinzip der KI-Sprachen.

Abb. 4:
Die Werte, die für eine Dame gespeichert werden, geben nicht etwa die Nummer der Spalte an, in der die Dame steht, sondern werden analog zu der in der Maske benutzten Repräsentation durch Bits benutzt.



Die Position einer Dame wird durch ein gesetztes Bit an der entsprechenden Stelle angezeigt. Das Verschieben der Dame erfolgt durch Shiften des Werts.
Durch Vergleich mit der Maske kann erkannt werden, ob die Position bereits besetzt ist.

Im hierzu genutzten Datentyp <unsigned> (vgl. Maske), wird das Bit gesetzt, das die Position in der Reihe kennzeichnet.
Dies läßt sich am einfachsten erreichen, indem man ein Bit des <unsigned> setzt und dieses durch 'shiften' über die einzelnen Positionen schiebt. Durch eine ODER-Verknüpfung von Maske und Damenposition läßt sich so leicht prüfen, ob die Dame in eine bereits besetzte Spalte gesetzt werden soll.

Natürlich muß, wie jede Rekursion, unsere Funktion auch ein Abbruchkriterium haben. Dieses ist erfüllt, wenn wir eine Dame in der letzten (achten) Reihe plaziert haben. In diesem Fall wurde eine Lösung gefunden. Statt eine weitere Rekursionsebene zu erzeugen, zählen wir den Zähler für die Lösungen um eins hoch und fahren mit dem Verschieben der Dame fort.

Wenn eine Dame in der letzten Spalte angekommen ist, haben wir alle Positionierungsmöglichkeiten geprüft. Jetzt kann der Funktionsaufruf beendet und die Rekursion auf der tieferen Ebene fortgesetzt werden (d.h. wir schieben die Dame in einer höheren Reihe weiter).

Hat die Funktion der ersten Rekursionsebene (erste Reihe) die Dame auf das letzte Feld gesetzt und die zugehörigen Möglichkeiten überprüft, so wird auch sie terminieren. Die Rekursion ist dann vollständig abgearbeitet, und unser Zähler enthält die Anzahl der Möglichkeiten. Das Hauptprogramm wird wieder aktiviert und kann diese Zahl ausgeben.

Mit der Rekursion haben wir den wichtigen Effekt, daß für jede Ebene (Reihe) eine neue Maske angelegt wird; wir müssen also beim Aufstieg auf eine höhere Reihe nicht den alten

Zustand wiederherstellen.

Wir haben durch die Rekursionsebenen sichergestellt, daß in jeder Reihe nur eine Dame stehen kann; die Maske garantiert uns, daß in jede Spalte nur eine Dame gesetzt wird; das letzte noch bleibende Problem ist das Überprüfen der Diagonalen.

Hierzu müssen wir prüfen, ob die zuletzt gesetzte Dame in einer Diagonalen mit den **vorher** gesetzten steht. Dazu kopieren wir uns einfach das Feldelement mit der Damenposition (wie aus Abb. 4 hervorgeht, kann man es als Maske für eine Reihe interpretieren) in zwei neue Masken, die mit rechts bzw. links bezeichnet werden. Diese beiden Masken werden pro Reihe, die wir nach oben laufen, jeweils um ein Bit nach links bzw. rechts verschoben und decken damit die Diagonalen zu der Dame ab (vgl. Abb. 5).

Abb. 5
Das Erzeugen der Diagonalen erfolgt, indem die Damenposition pro Reihe um eins nach links bzw. nach rechts geshiftet wird.

Wenn wir in einer der Diagonalen eine bereits gesetzte Dame finden, so können sich diese schlagen; es wird ein Fehlercode zurückgegeben. Sollten wir alle höheren Reihen überprüft haben, ohne auf eine Dame zu treffen, kann die neue Dame nicht geschlagen werden, es wird kein Fehlercode zurückgeliefert und der Rekursionsfunktion damit mitgeteilt, daß es sich lohnen kann, die tieferen Reihen (Rekursionsebenen) mit Damen zu besetzen.

Das so optimierte Programm hat eine Laufzeit von ca. 0.6 Sekunden, es fehlt uns allerdings noch der Vergleich zu den Benchmark-Tests der KI-Sprachen. Diese wurden für ein sechs-malsechs-Schachbrett mit sechs Damen gemacht; das abgedruckte Programm (Listing 1) berechnet die Lösungszahl für Schachbretter der Kantenlänge eins bis fünfzehn. Für ein sechs-mal-sechs-Brett habe ich bei meinen Tests Laufzeiten von etwa 0.04 Sekunden gemessen, selbstverständlich ohne Ausgabe der Lösungen (es kommt ja auf Zeit an). Damit ist wohl ein deutlicher Unterschied zu den KI-Sprachen zu sehen, ihre Werte lagen zwischen 13 und 130 Sekunden.
Das hergeleitete Programm nutzt dieselben Methoden wie die KISprachen (Rekursion); der Geschwindigkeitsunterschied entsteht aus der optimalen Abstimmung der Operationen und Datenstrukturen auf das Problem. Hierzu sind die KI-Sprachen nicht in der Lage, was aber nicht heißen soll, daß Probleme grundsätzlich besser auf herkömmliche Weise gelöst werden können.

Spielereien

Nachdem das Programm seinen Zweck nun erfüllt hat, können wir uns damit auseinandersetzen, dem Benutzer ein paar Bonbons für seine Aufmerksamkeit zu schenken. Als da wären: die Ausgabe der Lösungen oder sogar das bildliche Darstellen des "Damenwanderns".

Beide Probleme sind sehr einfach lösbar, sofern der Leser rudimentäre Kenntnisse in 'C' besitzt und bereit ist, den beschriebenen Lösungsweg am Programm nachzuvollziehen. Hierbei ist zu beachten, daß das abgedruckte Listing eine erweiterte Version für verschiedene Feldgrößen ist.

Die Ausgabe einer Möglichkeit kann erfolgen, sobald wir eine Lösung gefunden haben; es wird, statt die Lösungsanzahl hochzuzählen, eine Ausgaberoutine aufgerufen (s. Listing 2). Diese holt das Zählen der Möglichkeiten nach und gibt danach die Lösung auf dem Bildschirm aus; hierbei wird auf einen Tastendruck gewartet, um ein Betrachten des Bildschirms zu ermöglichen.

Die zweite Variante erfordert, daß jede Rekursionsebene ihre Reihe mit der entsprechend plazierten Dame selbst ausgibt. Die Ausgabe müßte also in der Schleife erfolgen, die die Dame über die Reihe schiebt. Wird die Rekursionsebene abgebrochen, so sollten die Felder der entsprechenden Reihe geleert werden, es gibt dann ja in dieser Reihe keine Dame mehr.

Thomas Pfoch

Listing 1:

```
 1:  /*****************************************/
 2:  /*                                       */
 3:  /*    Timerfunktionen fuer Benchmark-Tests */
 4:  /*                                       */
 5:  /*****************************************/
 6:
 7:  #include <osbind.h>
 8:  #include <stdio.h>
 9:
10:  typedef struct {
11:       long takte;
12:       long starttim;
13:  } timer;
14:
15:  #define Systime          *((long *)(0x466))
16:  #define tim_reset(tim)   (tim)->takte = (tim)
       ->starttim = 0L
17:  #define tim_read(tim)    ((tim)->takte)
18:
19:  static char *supstack;
20:
21:  tim_start(tim) timer *tim; {
22:
23:       supstack = (char *)Super(0L);
24:       tim->starttim = Systime;
25:       Super(supstack);
26:       return(tim->starttim);
27:  } /* timer start */
28:
29:  tim_stop(tim) timer *tim; {
30:       long      stime;
31:
32:       supstack = (char *)Super(0L);
33:       stime = Systime;
34:       tim->takte += stime - tim->starttim;
35:       Super(supstack);
36:       return(tim->takte);
37:  } /* timer stop */
38:
39:  /*****************************************/
40:  /* (c) Paris  n-Damen  Problem   1.7.87  */
41:  /*                                       */
42:  /*              Minimal-Version          */
43:  /*                                       */
44:  /* Dokumentation in ST 1/88; Da Felder in */
     /* C grundsaetzlich bei Null beginnen er- */
45:  /* haelt die erste Reihe den Index Null,  */
     /* jede tiefere ist als auch um eins      */
46:  /* kleiner.                              */
47:  /*****************************************/
48:
49:  #define MAX_FELD_GROESSE 15    /* zum Shiften
                  muss das high-bit Null sein */
50:
51:  int  Feldgroesse;
52:  int  Spaltenmaske;
53:  long Anzahl;
54:
55:  Setze_Dame(Feld,Reihe,Maske) int *Feld;
       register int Reihe; int Maske; {
56:     int  Spalte = 1,
          lokalfeld[MAX_FELD_GROESSE];
57:     register  int  k, links, rechts, *feldP,
          *lokalP;
58:
59:     for (Spalte=1;Spalte<=Spaltenmaske;
          Spalte<<=1){
60:          if (Maske & Spalte) continue;
61:          feldP = Feld + Reihe;
62:          if (*feldP & Spalte) continue;
63:          if (Reihe >= Feldgroesse-1) Anzahl++;
64:          else {
65:               lokalP = lokalfeld + Reihe;
66:               links = rechts = Spalte;
67:               for (k=Reihe;k<Feldgroesse;k++) {
68:                    links >>= 1;
69:                    rechts <<= 1;
70:                    *(++lokalP) = (*(++feldP)
                         | links
                         | rechts) & Spaltenmaske;
71:               }
72:               if (lokalfeld[Reihe+1] !=
                    Spaltenmaske)
73:                    Setze_Dame(lokalfeld,Reihe+1,
                         (Maske|Spalte));
74:          }
75:     }
76:  } /* Setze eine Dame in die Reihe[Reihe] */
77:
78:  main() {
79:       int i, Feld[MAX_FELD_GROESSE];
80:       timer tim;
81:
82:       Spaltenmaske = 0;
83:       for (Feldgroesse=1; Feldgroesse<=
            MAX_FELD_GROESSE; Feldgroesse++) {
84:            for (i=0;i<Feldgroesse;i++) Feld[i] = 0;
85:            Anzahl = 0L;
86:            Spaltenmaske = (Spaltenmaske << 1) + 1;
87:              tim_reset(&tim);
88:              tim_start(&tim);
89:            Setze_Dame(Feld,0,0);
90:              tim_stop(&tim);
91:            printf("Feldgr =%3d: %6ld Loesungen,
                 VBL-Takte: %-8ld(%-7ld sec)\n",
92:                 Feldgroesse,Anzahl,tim_read(&tim),
                      tim_read(&tim)/71L);
93:            fflush(stdin); getchar();
94:       }
95:       fflush(stdin); getchar();
96:  } /* Hauptprogramm */
```

Listing 2:

```
 1:  /*****************************************/
 2:  /* (c) Paris  acht-Dame Problem  1.7.87  */
 3:  /*                                       */
 4:  /*              Minimal-Version          */
 5:  /*                                       */
 6:  /* Dokumentation in ST 1/88; Da Felder in */
     /* C grundsaetzlich bei Null beginnen er- */
 7:  /* haelt die erste Reihe den Index Null,  */
     /* jede tiefere ist als auch um eins      */
 8:  /* kleiner.                              */
 9:  /*****************************************/
10:
11:  #include <stdio.h>
12:
13:  int  Feld[8];
14:  long Anzahl;
15:
16:  Setze_Dame(Reihe,Maske) register int Reihe,
       Maske; {
17:
18:    register int Spalte = 1;
19:
20:    for (Spalte=1;Spalte<=0xFF;Spalte<<=1) {
         /* fuer alle Spalten der Reihe */
21:      if (Maske & Spalte) continue;
           /* weiter, wenn Spalte besetzt */
22:      Feld[Reihe] = Spalte; /* sonst 'merken'*/
23:      if (diagonal(Reihe)) continue; /* wenn
           schlagbar naechste Sp. */
24:      if (Reihe >= 7)  Anzahl++; /* wenn achte
           Reihe => Loesung */
25:      else Setze_Dame(Reihe+1,(Maske | Spalte))
           ; /* sonst naechste Reihe setzen */
26:    }
27:  } /* Setze eine Dame in die Reihe[Reihe] */
28:
29:  diagonal(Reihe) register int Reihe; {
30:
31:    register int i, *Testreihe, links, rechts;
32:
33:    Testreihe = Feld + Reihe;
34:    links = rechts = *Testreihe--;
35:    for (i=1;i<=Reihe;i++,Testreihe--) {
         /* rueckwaerts ueber alle Reihen */
36:      links  <<= 1; /* Diagonalenmaske fuer
           linke    */
37:      rechts >>= 1; /* und rechte Diagonale
           bilden.  */
38:      if ((*Testreihe & links) || /* wenn
           Dame in der Diag. steht  */
39:         (*Testreihe & rechts)) /* koennen
              sich beide schlagen!  */
40:           return(1); /* Rueckgabe des
                Fehlers. */
41:    }
42:    return(0); /* keine Dame in den
         Diagonalen  */
43:      /* OK => weitere Reihen besetzen */
44:  } /* Teste die linke und rechte Diagonale */
45:
46:  main() {
47:
48:    Setze_Dame(0,0); /* Rekursion aktivieren */
49:    printf("Es gibt %ld Loesungen\n",Anzahl);
         /* Ausgabe der Loesung      */
50:    fflush(stdin); getchar(); /* auf
         Bestaetigung warten */
51:  } /* Hauptprogramm */
```

Sprit(e)zige Grafiken

Je leistungsfähiger und fortschrittlicher die heutigen Rechner sind, desto höher sind die Erwartungen bezüglich der Grafik. Der ST liefert eine für Animationen und Spiele recht ansprechende Farbvielfalt und Grafikauflösung. Was liegt näher, als ein professionelles Programm zur Gestaltung und Animation von Sprites, eben den PRO SPRITE DESIGNER, auf den Markt zu bringen? Wir wollen diesen Sprite-Editor näher unter die Lupe nehmen.

Die allgemeine Aufmachung des Programms zeigt schon in der Verpackung ihren Charakter: passend zu dem Label der Firma TRIANGEL wird das Programm in einer dreieckigen, gut entworfenen Verpackung geliefert. In ihr befinden sich eine Programmdiskette sowie eine Anleitung, die allerdings durch ihr kleines Format (Diskettengröße) etwas unangenehm auffällt, was glücklicherweise nichts über deren Inhalt aussagt, denn dieser ist ausführlich und ansprechend gestaltet.

PRO SPRITE DESIGNER ist ein Programm, mit dem man SPRITES in niedriger Auflösung erstellen und, was äußerst wichtig ist, in eigenen Programmen verwenden kann, denn ein Erstellen von Grafiken, ohne daß Sie verwendbar sind – einige Programme auf dem Markt zeugen von solchem Unsinn –, ist sicherlich vergebene 'Liebesmüh'. Der DESIGNER bietet die Möglichkeit dazu, die Datenmengen in verschiedenen Programmiersprachen auszugeben, wie zum Beispiel ASSEMBLER, C und BASIC. Die Größe eines Sprites beläuft sich auf 32x22 Punkte, wobei es eine sehr einfache Möglichkeit gibt, Sprites mit der doppelten Höhe, also mit 32x44 Punkten, zu erstellen.

Laßt die Bilder laufen

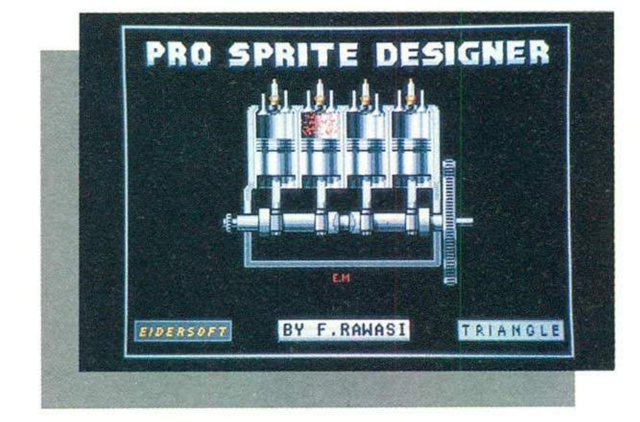


Bild 1: Ein Bild aus Motor-Demonstration

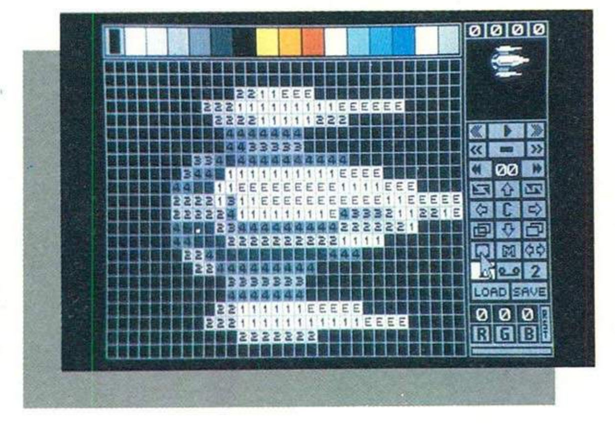


Bild 2: Editieren mit einer besonderen Markierung des Sprites

Die wahrscheinlich interessanteste Möglichkeit, die PRO SPRITE DESIGNER zur Verfügung stellt, ist das Erstellen von Bildfolgen, sogenannten Animationssequenzen. Damit ist es optimal geeignet, bewegte Sprites zu erzeugen, die später in Spielen Anwendung finden sollen. Aber nicht nur dafür sind animierte Sprites gefragt. Als Beispiel befindet sich eine schön gestaltete Demonstration eines Benzinmotors auf der Diskette, die die Möglichkeiten von PRO SPRITE DESIGNER überzeugend darstellt.

Bei der Gestaltung von PRO SPRITE DESIGNER ist darauf geachtet worden, das Programm möglichst einfach handhabbar zu machen, was sich in dem durch Piktogramme erstellten Kommandotableau zeigt (Bild 2). Die sehr ausführliche Anleitung verleiht dem vielfältigen und zunächst verwirrend aussehenden Tableau erhält durch sehr viel an Übersichtlichkeit, so daß ein vernünftiges Arbeiten mit diesem Editor innerhalb kurzer Zeit möglich ist. Beim Editieren fehlt kein Veschieben und Spiegeln der Sprites, auch

deren Vervielfältigung ist einfach möglich. Das Erstellen einer Animationssequenz funktioniert beispielsweise, indem man einige Sprites zeichnet, Anfang und Ende markiert und diesen 'Film' ablaufen läßt, wobei die Geschwindigkeit natürlich einstellbar ist. Es besteht sogar die Möglichkeit, die Farben im Sprite beim Editieren mit einer besonderen Markierung zu versehen, so daß deren Zuordnung zur Farbpalette sehr erleichtert wird.

Einbau leicht gemacht

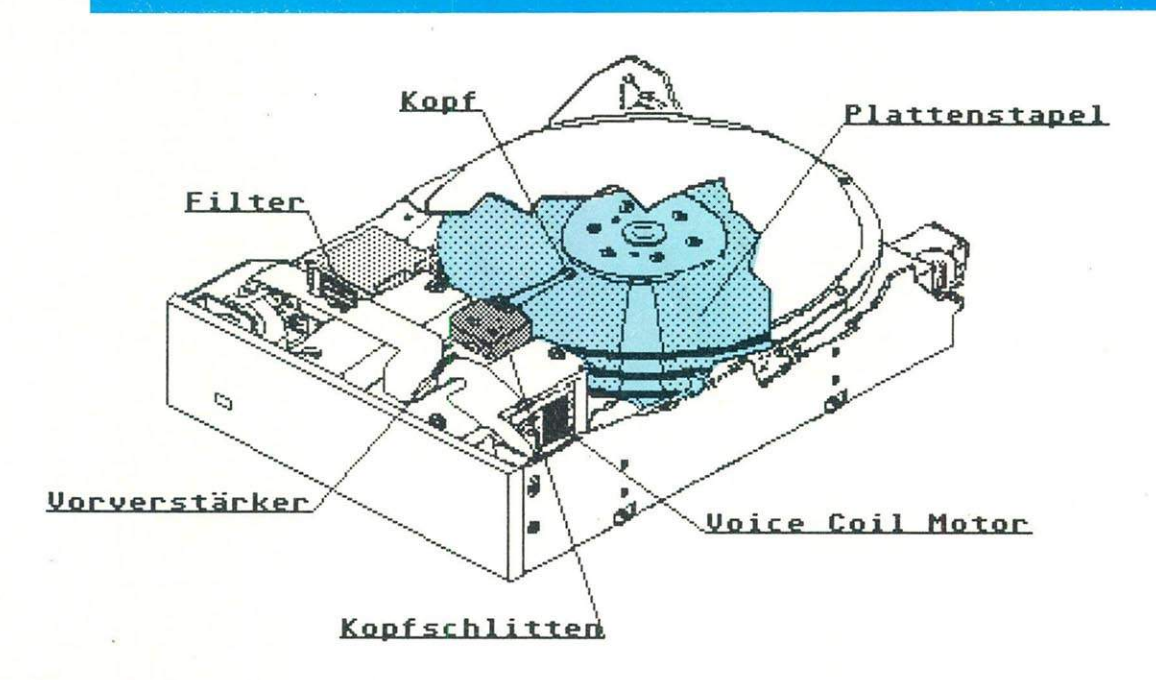
Der eigentliche Sinn der Kreation einzelner Sprites oder Sprite-Sequenzen ist deren Benutzung in eigenen Programmen. Dazu wird zum Programm eine Ansammlung von Maschinenroutinen mitgeliefert, die verhältnismäßig einfach in eigene Programme eingebaut werden können. Durch ein mitgeliefertes Zusatzprogramm können die von PRO SPRITE DESIGNER ausgegebenen Daten in (programmiersprachen-) verständlichen Daten ausgegeben werden. So werden für BASIC DataZeilen und für 'C' Arrays erzeugt. Wer diese Daten nicht direkt im Programm haben möchte, kann sie selbstverständlich auch erst nach Programmstart einlesen und der Maschinenroutine zum Zeichnen der Sprites übergeben.

Alles in allem stellt PRO SPRITE DESIGNER ein ansprechendes, gut bedienbares und nützliches Programm dar. Sieht man davon ab, daß es nur in der niedrigsten Auflösung arbeitet – wer schreibt aber schon Spiele, die in der mittleren Auflösung Sprites benutzen? – kann man diesen Editor als gutes Werkzeug zum Erstellen von Sprites anerkennen. Einziger Kritikpunkt ist die zwar schöne, aber recht unpraktische Verpackung, deren Haltbarkeit und Form sicher nicht den Wünschen professioneller Programmierer entsprechen; sieht man doch, daß es beim PRO SOUND DESIGNER der gleichen Firma, der in einem Ring-Ordner ausgeliefert wird, anders und trotzdem ansprechend zu machen war. Dies wird aber einen Interessenten sicherlich nicht vom Kauf des Programms abhalten können.

(SH)

EIDERSOFT DEUTSCHLAND
CCD D. Beyelstein
Burgstr. 9
D-6228 Eltville

Die Festplatte



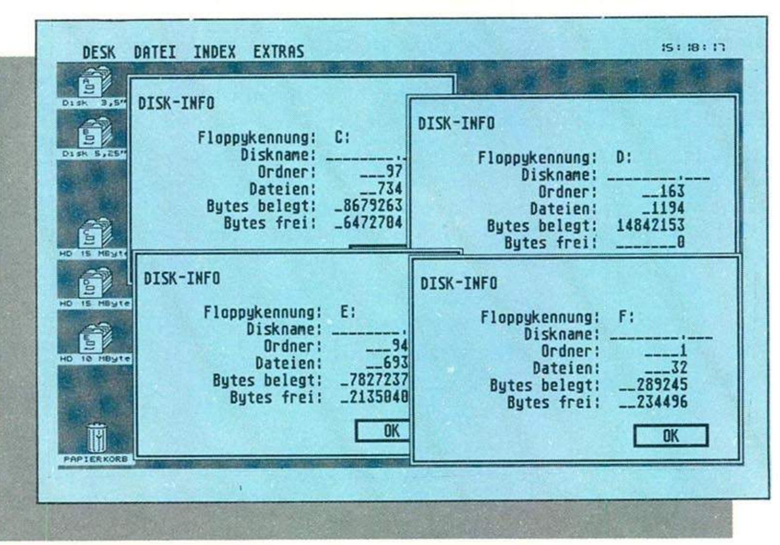
Während über die Floppy am ATARI ST inzwischen schon viel Information zur Verfügung steht, ist es um die Festplatte SH 205 bis auf ein paar kurze Testberichte doch recht ruhig geblieben. Ich möchte Sie deshalb einladen, Ihren Platz an der Tafel einzunehmen, und mit mir in dieser Artikelserie ein bißchen von der "Festplatte" zu naschen...

Nachdem ich von einem Bekannten ein 40 MByte-Laufwerk erstanden hatte (zu einem Preis, bei dem ich nicht "Nein danke" sagen konnte), kamen mir doch erste Zweifel, ob ein Umbau meiner SH 205 überhaupt möglich sein würde. Auch das Fehlen von Informationen zu diesem Thema machte ihn nicht gerade leichter. Das Resultat des Umbaus können Sie im nebenstehenden Bild sehen (Wo ist denn nur der Taschenrechner ?...).

Ich möchte Ihnen im Laufe dieser Serie die Informationen und Daten zum Thema Festplatte (soweit sie überhaupt zur Verfügung stehen) vermitteln und auch meinen Umbau detailliert beschreiben, vielleicht haben Sie ja auch einen Bekannten, der... (im übrigen sind die Preise für 40-MByte Laufwerke teilweise schon unter DM 1.000,- gefallen). Ich beginne in dieser Folge mit der allgemeinen Beschreibung eines Festplattenlaufwerks, das nächste mal ist der Controller dran, danach folgt das Zerlegen einer SH 205 in ihre Einzelteile (keine Angst – nur im Text). Auch wenn Sie nicht an einen Umbau denken, vielleicht ist für Sie doch die eine oder andere Information dabei.

Es war einmal...

...so um die Mitte der 70er Jahre, als die ersten Festplatten entwickelt wurden. Der in diesem Zusammenhang oft genannte Name 'Winchester Drive' hat nichts mit dem bekannten Mehrlader-Gewehr gemeinsam, es war damals ein Projektname von IBM, die in ihrem 'Winchester-Labatory' einen Kopfträger für Festplatten entwickelt hatten (Heute wird in der Regel die sogenannte Whit-



ney Technology verwendet). Die Laufwerke waren erst mit 14" (ca. 35 cm), dann mit 8" (ca. 20 cm) großen Platten ausgestattet. Diese Laufwerke waren groß, schwer und teuer und hatten mit den heute erhältlichen nur die prinzipielle Funktion gemeinsam. Erst 1980 wurde von der Firma Seagate ein 5 1/4"-Laufwerk mit umwerfenden 5 MByte vorgestellt; die danach rasant einsetzende Technologieverbesserung erreichte inzwischen Plattengrößen herab bis zu 3 1/2" und Kapazitäten bis knapp 400 MByte. (An dieser Stelle sei mir für die Megabyte-Verwöhnten ein kleiner Nostalgieanflug erlaubt; Mein erster Mikrocomputer, ein PET – auch von Herrn Tramiel –, hatte tierische 8 k RAM; die für DM 1.300,- gekaufte Speichererweiterung von 24 k RAM war natürlich reiner Größenwahn. Welche Programme sollten denn je 32 k belegen ?!...)

Hart oder weich...

Das sollte keine Frage nach Ihren Frühstücksgewohnheiten sein, sondern beschreibt schon den prinzipiellen Unterschied zwischen 'Floppy' und 'Festplatte'. Im Gegensatz zur Floppy (also weichen) Disk finden wir in der Festplatte (wie der Name eben schon sagt) harte Scheiben aus Aluminium (wegen des Gewichts) vor, auf denen die eigentliche magnetische Schicht aufgebracht ist. Noch kurz etwas zu den unterschiedlichen Namen für Festplatten, als da wären:

- Winchester Drive,
- Harddisk,
- Fixed Disk

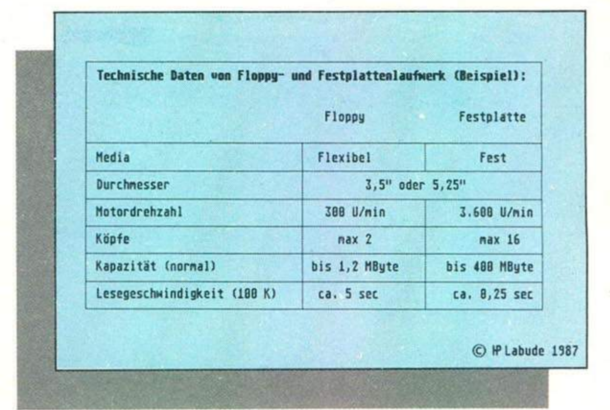
Sie beschreiben eigentlich alle dasselbe: 'Fixed Disk' ist z.B. die offizielle Bezeichnung dieser Platten von IBM im Gegensatz zur 'Floppy Disk' – eben weil sie im Laufwerk fixiert sind, was aber auch nicht stimmt, weil es ja Festplattenlaufwerke mit wechselbaren Platten gibt – diese sind halt nur fest; aber halt: Gibt es da nicht die neue Sandwich-Methode von 3M, die flexible Platten für Festplattenlaufwerke... Lassen wir das. Wir einigen uns darauf, daß mit Harddisk's – pardon, Festplatten – das komplette Laufwerk gemeint ist, welches ich nun näher beschreiben möchte.

Jetzt geht's los...

Auf der Festplatte lassen sich, wie auf der Floppy, Daten speichern, wieder abrufen und auch löschen. Was die Festplatte dabei so interessant macht, ist einmal die wesentlich größere Kapazität wie auch der deutlich schnellere Zugriff (ein Beispiel dafür finden Sie in der nebenstehenden Tabelle). Einen Grund dafür hört man, wenn eine solche Festplatte eingeschaltet wird. Während die Floppy in der Regel mit 300 Umdrehungen pro Minute arbeitet, können es bei einer Festplatte über das zehnfache, nämlich 3.600 U/min pro Minute sein (Es gibt auch kleinere und größere Umdrehungszahlen, ich will mich bei diesen Angaben aber schon jetzt an die technischen Daten halten, wie sie auch für die SH 205 zutreffen).

Take Off

Wenn Sie mal nachrechnen wollen: Bei dieser Umdrehungszahl ergibt sich bei einer 5 1/4"-Platte im Randbereich eine Geschwindigkeit von fast 90 km/h, mit der sich die Scheibe unter dem Kopf bewegt. Betrachtet man die glänzende Plattenoberfläche unter dem Mikroskop, so erkennt man, daß diese doch nicht so glatt ist, sondern eher dem Asphaltbelag einer Straße entspricht. Versuchen Sie doch mal, mit einem Auto ohne Räder auf 90 km/h zu kommen... Was bei dem Floppylaufwerk noch funktioniert – dort liegt der Kopf auf der Diskette auf (auch hier gibt es mit der Zeit einen gewissen Abrieb – weshalb werden sonst so viele Reinigungssets für Floppylaufwerke angeboten ?!) ist bei den Umdrehungszahlen einer Festplatte nicht mehr möglich – ein aufliegender Kopf würde als Schleifklotz fungieren. Wie aber den Kopf von der Platte hochbekommen? Eine Möglichkeit wäre, über die Kopfträger einen konstanten Abstand zu halten – bei allem Respekt vor der Feinmechanik wäre das doch ein zu schwieriges Unterfangen. Der Abstand wäre viel zu groß, um eine intensive Magnetisierung der Schicht

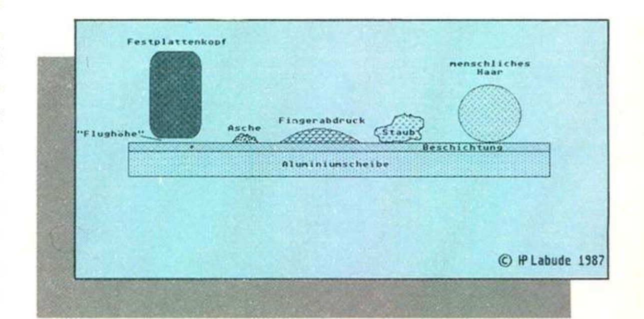


Technische Daten von Floppy- und Festplattenlaufwerk (Beispiel):	Floppy	Festplatte
Media	Flexibel	Fest
Durchmesser	3,5" oder 5,25"	
Motordrehzahl	300 U/min	3.600 U/min
Köpfe	max 2	max 16
Kapazität (normal)	bis 1,2 MByte	bis 400 MByte
Lesegeschwindigkeit (100 K)	ca. 5 sec	ca. 0,25 sec

© HP Labude 1987

zu erreichen, außerdem wäre das Aufzeichnungsformat viel zu grob, die benachbarten Spuren lägen zu weit auseinander. Der Kopf (der geistige), der sich nun einen Trick einfallen ließ, muß wohl begeisterter Flieger sein, denn genau dies machen die Köpfe einer Festplatte – sie fliegen. Durch die Drehung der Scheibe wird aufgrund der Reibung an der Oberfläche der Platte die Luft in Bewegung versetzt. Da die Köpfe zudem noch aerodyna misch geformt sind, reicht die hiermit gewonnene Auftriebskraft zum 'Take Off' für die paar Gramm Gewicht von Kopf und Kopfträger aus.

Durch einen gewissen Anpreßdruck des Kopfträgers ergibt sich jetzt eine relativ konstante Entfernung – oder sagen wir lieber Nähe zur Platte, denn bei den heutigen Laufwerken schwebt der Kopf etwa 0,00025 mm über der Plattenoberfläche. Eine Vorstellung von den Größenverhältnissen bekommen Sie beim Betrachten des nebenstehenden Bildes. Dort können Sie auch den



Grund sehen, warum sich Ihr Händler weigert, ein solches Laufwerk zu öffnen, z.B. um die Anzahl der Platten nachzu-

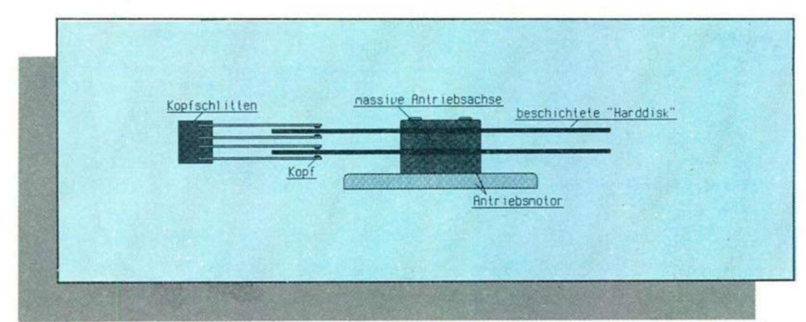
zählen (Sollte er es doch tun, wechseln Sie lieber den Händler...). Damit kein Staub oder sonstige Verunreinigungen den Flug unseres Kopfes behindern, werden die Festplattenlaufwerke ausschließlich in Cleanrooms montiert und hermetisch versiegelt. (Darum sind Reparaturen an beschädigten Platten auch so teuer, nicht jeder hat einen solchen Raum zur Verfügung. In der BR Deutschland existiert meines Wissens ein einziger in München!).

Runter kommen sie immer...

Leider läßt sich beim Betrieb nicht verhindern, daß hie und da Verunreinigungen auftreten. Nach dem Abschalten der Versorgungsspannung läuft der Motor bewußt langsam aus, um den Köpfen eine weiche Landung zu ermöglichen. Dies geschieht entweder irgendwo (nicht gut) oder definiert auf einer sogenannten Parkspur auf der Platte selbst (schon besser). Hochwertige Laufwerke fahren die Köpfe sogar in eine spezielle Halterung außerhalb der Platte (sehr gut). Bei diesen Landungen kommt es jetzt auf jeden Fall zu einem gewissen Abrieb (Keine Angst, die Hersteller garantieren heute 10.000 und mehr Landungen ohne einen anschließenden Defekt zu produzieren). Damit dieser Dreck jetzt nicht irgendwo in der Festplatte herumirrt oder sich womöglich auf der Oberfläche niederläßt (s.o. – oder stellen Sie sich vor, Sie fahren mit 90 km/h auf der Landstraße, und der vor Ihnen auf der Fahrbahn auftauchende Felsblock paßt leider doch nicht unter Ihrem Wagen durch...), haben sich die Herren Ingenieure wieder etwas Feines einfallen lassen. In dem Laufwerk ist ein Mikrofilter eingebaut, welcher Partikel <0,3 ym zu fast 100% herausfiltert. Damit die Luft (und damit der Dreck) auf jeden Fall durch den Filter marschiert, wird der Effekt der Luftbewegung durch die Platten ausgenutzt (ähnlich wie bei einem Tangentiallüfter); die spezielle Form des umgebenden Rahmens bestimmt dabei die Richtung des Luftstromes und drückt diesen durch den Filter.

Cool bleiben...

Die Aluminiumplatte mit den magnetischen Schichten ist auf einer massiven Achse befestigt. Diese wird wiederum in einem Präzisionslager geführt und



von einem Motor im Inneren der Achse direkt angetrieben. Dieser verhältnismäßig hohe Aufwand, verbunden mit einem massiven Gehäuse, soll verhindern, daß Temperaturschwankungen den Betrieb des Laufwerks beeinflussen. Deshalb sollten die Arbeitstemperaturen der Festplatten tunlichst eingehalten werden. Sie können sich ja leicht vorstellen, was Verformungen des Gehäuses oder des Lagers für Auswirkungen haben können. Da dieser relativ hohe mechanische Aufwand bei jedem Festplattenlaufwerk notwendig ist, sind Bestrebungen, die Kapazität weiter zu erhöhen, verständlich. Eine Möglichkeit bezieht sich auf die Platten selber. Die ersten Magnetschichten enthielten Metalloxid, dessen Kristalle (magnetische Dipole) nadelförmig waren und sich bei der Beschichtung beliebig ausrichteten. Dadurch wurde eine relativ "dicke" Schicht notwendig (etwa 0,001 mm stark). Für die Information mußten recht viele benachbarte Kristalle magnetisiert werden, was wiederum zu breiten Abständen benachbarter Informationsbits führte. Erst mit der Einführung des sogenannten Plated Media (das ist die Bezeichnung für diese speziell beschichteten Scheiben) konnten die Abstände deutlich verringert (und damit die Anzahl der Bits pro Spur erhöht) werden. Ohne auf die aufwendigen technischen Einzelheiten einzugehen (Beschuß des Metalloxids durch Edelgas), ist das Resultat eine bei der Produktion exakt zu bestimmende Richtung der magnetischen Teilchen und damit eine geringe Stärke der Magnetschicht. Eine dünnere Schicht (nur noch 0,0001 mm) braucht aber nicht so stark magnetisiert zu werden und liefert trotzdem genauere Informationen.

Köpfchen, Köpfchen...

Bei den alten Köpfen der Winchester-Technologie werden kleine Spulen verwendet, die um einen Kern mit einem winzigen Spalt gewickelt sind. An diesem Spalt entsteht bei einem Stromfluß in der Spule ein Magnetfeld, welches wiederum die darunterliegende Schicht (der Platte) magnetisiert. Umgekehrt induziert die Information auf der Platte eine Spannung in der Spule. Wie wir in der Schule gelernt haben, erfolgt bei einer Induktion auch eine unschöne Gegeninduktion, die einen schnellen Wechsel des magnetischen Feldes verhindert. Damit ist die Anzahl der Bits auf der Platte beschränkt, wir müssen ja warten, bis das Magnetfeld sich beruhigt hat. Da sich in dieser Zeit die Scheibe aber weiter dreht, verlieren wir kostbare Informationsmengen. Hier hat die Dünnfilmkopftechnik einen Ausweg gebracht. Dieser Kopf besteht aus Permalloy (spezielle Legierung, auch bekannt aus der HiFiTechnik) und kann sein magnetisches Feld zehnmal öfter als eine Spule wechseln. Außerdem ist das erzeugte magnetische Feld wesentlich exakter und ermöglicht ein engeres Beschreiben einer Spur.

Hochstapelei...

In den heutigen 5 1/4"-Festplatten gibt es inzwischen bis zu 8 Platten (allerdings nur bei vol-

ler Bauhöhe, in der SH 205 finden wir nur Laufwerke mit halber Höhe). Obwohl es im ersten Moment paradox klingt, sind diese Festplatten automatisch schneller im Zugriff. Warum das so ist? Nun, jede Plattenseite muß logischerweise mit einem eigenen Kopf abgetastet werden. Die Kopfträger sind alle gemeinsam auf einem 'Schlitten' befestigt, welcher von einem Motor bewegt wird. Und genau dies wird ausgenutzt. Wir versehen nur die oberste Platte (jetzt nur theoretisch zum besseren Verständnis) mit der Spurnummer. Als weitere Angabe brauchen wir jetzt nur noch die Anzahl der Köpfe. Dadurch daß die Köpfe ständig übereinander stehen, werden die gleichzeitig durch die Köpfe abgetasteten Spuren (auf jeder Platte und Seite) zu einem sogenannten 'Zylinder' zusammengefaßt. Rein rechnerisch bedeutet das

Byte pro Spur x Anzahl der Köpfe = Byte pro Zylinder

Hat also eine Spur (die wiederum wie bei der Diskette in Sektoren eingeteilt ist) z.B. 8.704 Byte, so hat ein Zylinder in einem Laufwerk mit vier Köpfen die Kapazität von 34.816 Byte. Wieso das schneller sein soll? Ganz einfach – um nun ein File mit einer Größe von 68 kByte zu schreiben, braucht ein solches Laufwerk den Kopfschlitten gerade um eine Spur weiter zu bewegen (Die Elektronik versucht immer erst einen Zylinder vollzuschreiben, bevor sie einen Spurwechsel veranlaßt). Handelte es sich um eine einseitige Diskette, wären bei dem gleichen File

512 Byte * 9 Sektoren = 4.608 Byte pro Spur

68 kByte / 4.608 BpS = ca. 15 Spuren

notwendig. Hier erkennt man schon einen der Gründe, warum Festplatten schneller sind, denn ein Spurwechsel kostet Zeit (in der Größenordnung von Millisekunden – damit wir uns da nicht falsch verstehen...).

Schritt für Schritt

Für die Bewegung des Kopfschlittens gibt es zwei gängige Antriebstypen. Der erste ist ein Schrittmotor, wie er auch im Diskettenlaufwerk Verwendung findet. Es ist ein robuster Antrieb, wobei die schrittweise erfolgende Drehbewegung über ein Metallband an den Kopf übertragen wird. Die Ansteuerung dieses Motors ist relativ 'einfach', da nur mit Stepimpulsen gearbeitet werden muß. Der jeweilige Drehwinkel wird von dem im Motor vorhandenen Magnetfeld auch ohne angelegte Betriebsspannung gehalten.

Nachteil dieser Antriebsart ist die etwas höhere Spurwechselzeit durch die notwendige Pause zur Beruhigung des Kopfes (ähnlich einer Kugel, die in eine Senke läuft – es dauert einen Moment, bis sie an der tiefsten Stelle zur Ruhe kommt). Laufwerke mit Schrittmotor kommen auf mittlere Zugriffszeiten zwischen 40 und 85 ms.

Die zweite Antriebsart ist der sogenannte Linearmotor (Voice Coil Motor). Ähnlich einem Lautsprecher wird hier die Kraft von einer Spule erzeugt, die sich in einem konstanten Magnetfeld befindet. Je größer der Strom, desto größer die Kraft und damit die Auslenkung. Damit die Elektronik jetzt weiß, wo genau sich die Spule befindet, gibt es einen Servokreis, in dem durch das Magnetfeld wiederum eine Spannung erzeugt wird. Diese wertet die Elektronik aus. Ein anderer Weg scheint auf den ersten Blick etwas verschwenderisch zu sein, hat aber durchaus rationale Gründe. Der elektronische Regelaufwand ist sehr groß und teuer. Da man diese Antriebsart normalerweise nur in Laufwerke mit hohen Kapazitäten einbaut, 'requiriert' der Hersteller einfach eine Seite aus dem Plattenstapel. Dort wird bei der Herstellung eine Spurinformation aufgezeichnet, anhand deren die Regelelektronik sofort weiß, in welcher Position die Köpfe sich gerade befinden. In beiden Fällen ergeben sich aber geringe Spurwechselzeiten zwischen 5 und 40 ms (Diese Angaben sind mittlere Zugriffszeiten. Es ist klar, wenn der Kopf gerade mal auf die benachbarte Spur 'hüppen' muß, geht das schneller, nur falls er halt doch ganz auf die andere Seite muß...). Ein scheinbarer Nachteil – dieser Motor (die Spule) muß ständig von Strom durchflossen sein – wird für den schlampigen Benutzer der Festplatte (ehrlich, wer fährt den schon den Kopf vor dem Abschalten in die Parkspur ?!...) zum Vorteil. Im Moment des Abschaltens wird entweder über eine Feder oder eine Resternergie (Kondensator oder Umpolen des Plattenmotors zum Generator) der Kopfschlitten in eine definierte Position gefahren und dort verriegelt (!). Ohne Strom würden sonst die "Voice Coil" und damit die Köpfe in der Gegend "rumschlabbern".

Bit für Bit

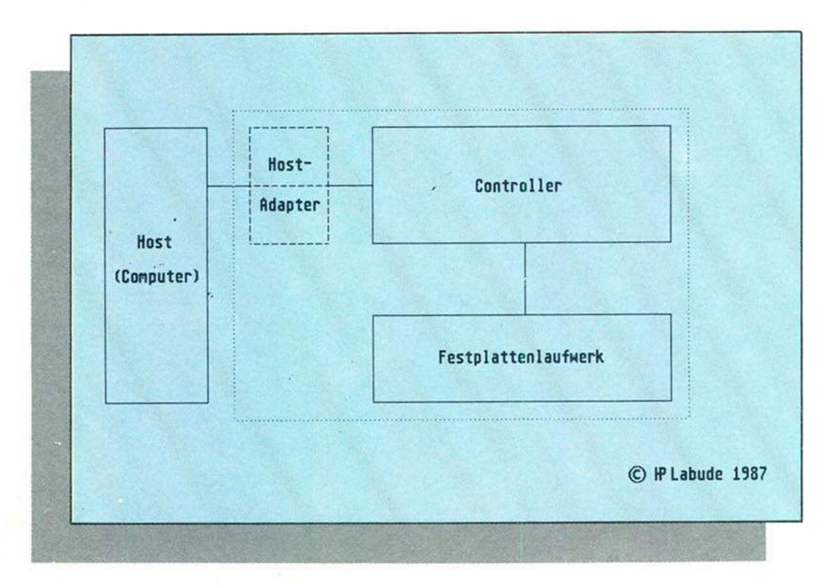
Ich will Sie jetzt nicht zum Trinken auffordern, wir wenden uns nur kurz der Aufzeichnungselektronik zu. Am Laufwerk befindet sich ja noch eine große Platine, auf der allerlei Elektronik zu sehen ist. Diese hat aber normalerweise nur die niederen Arbeiten wie Kontrolle der Drehzahl und Steuerung des Kopfantriebes zu erledigen (und noch ein paar andere Sachen, dazu kommen wir beim nächsten Mal). Einen zweiten Teil der Elektronik können Sie sowieso nicht sehen, der befindet sich nämlich im Laufwerk selber. Die durch die Platteninformation in die Köpfe induzierte Spannung ist so gering, daß die ersten Vorverstärker schon auf dem Kopfschlitten sitzen. Halten Sie sich dabei bitte die gewaltigen Datenmengen und Geschwindigkeiten vor Augen. Die Transferrate liegt bei den in der SH 205 verwendeten Laufwerken bei 5 MBit/s (!). Damit ist gleichzeitig auch die Kapazität einer Spur festgelegt:

Die Platte dreht sich 60 mal in der Sekunde unter dem Kopf, d.h.

5 MBit / 60 = 83.333 Bit pro Spur

das sind ca. 10,2 kByte pro Spur. Nehmen wir einmal die

gleiche Bytezahl pro Sektor, wie wir sie bei der Diskette haben (512 Datenbytes plus ein paar Bytes zum Synchronisieren), so kommen wir auf etwa 17 – 18 Sektoren pro Spur. (Diese Angabe gilt für das MFM-Aufzeichnungsverfahren, wie wir es schon von der Floppy kennen. Da dieses Thema aber stärker mit dem Controller als dem Laufwerk zusammenhängt, möchte ich Sie schon jetzt auf die zweite Folge der Artikelserie verweisen.)



Erklärung der Abkürzungen (alphabetisch):

ACSI = ATARI Computer System Interface
ATARI-Bezeichnung für den DMA(Direct Memory Access)-Port

ANSI = American National Standard Institute
Normeninstitut in Amerika, ähnlich unser Deutschen Industrie Norm-Gesellschaft

ESDI = Enhanced Small Device Interface
neuere Festplattenschnittstelle, die unglücklicherweise die gleiche Steckkontakte wie die ST 506 hat, aber ansonsten grundverschieden belegt ist. Dreifach schnellere (serielle) Datenübertragung als bei der ST 506, dadurch dreifache Festplattenkapazität möglich.

SASI = Shugart Accociates System Interface
eine von Shugart entwickelte Schnittstelle für Winchester-Laufwerke; Vorläufer der SCSI-Norm

SCSI = Small Computer System Interface
Von der ANSI akzeptierte Standardschnittstelle für parallele Datenübertragung (8-Bit)

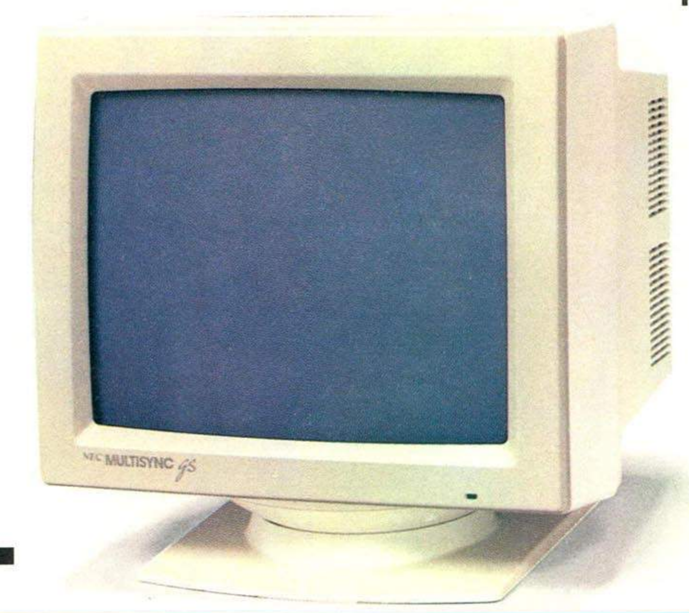
ST506 = Seagate Technologies
Das ST 506 war meines Wissen das erste Laufwerk von Seagate mit 5 MByte Kapazität, die hierzu entwickelte Schnittstelle (ähnlich der Shugart-Schnittstelle für Floppylaufwerke) wurde einfach als Standard übernommen.

Verbindung

Das Festplattenlaufwerk alleine ist genauso "dumm" wie ein Floppylaufwerk, es braucht zum Betrieb seinen Kontrolleur (wie gesagt, davon mehr im nächsten Heft) und dieser wiederum ja Daten, die er auf der Festplatte speichern kann. Damit das alles funktioniert, müssen die einzelnen Teile verbunden sein. Und da tauchen sie dann auf – die Schnittstellen (in Neudeutsch: Interface). Betrachten Sie sich im nebenstehenden Schaubild bitte mal den grundsätzlichen Aufbau eines Festplattensystems. Sie erkennen, daß wir mindestens zwei Schnittstellen brauchen (stimmt nicht s.u.), um die Teile zu verbinden – einmal zwischen Computer und Controller und zum zweiten zwischen Controller und Laufwerk. Und da finden sie sich dann, die SASI, SCSI, ACSI, ST506, EDSI und wie sie alle heißen. In der nebenstehenden Tabelle finden Sie die ersten Entschlüsselungen für dieses babylonische Kürzelwirrwarr. Es gibt jetzt allerdings Laufwerke, welche z.B. einen SCSI-Controller gleich eingebaut haben. Funktionell besitzen sie aber den gleichen Aufbau wie ein Laufwerk mit externem Controller. Oder die Computer (das sind die sog. Host) besitzen keinen genormten Anschluß (z.B. DMA-Port von ATARI), dann muß noch ein Host-Adapter eingefügt werden (siehe SH 205). Welche technischen Unterschiede die einzelnen Schnittstellen haben, davon beim nächsten Mal mehr.

HPL

Vor etwa einem Jahr stellten wir den damals noch neuen MultiSync Farbmonitor von NEC vor und erklärten, wie man sich leicht



Schwarz auf Weiß –

Der neue MultiSync-Monitor

ein Anschlußkabel für den ATARI ST bauen kann. Mittlerweile gehört dieser Monitor zumindest im PC Bereich zu den Marktführern.

Doch was ist das Besondere an ihm? Es gibt ja schließlich jede Menge Farbmonitore und das wesentlich kostengünstiger. Der MultiSync aber ermöglicht alle drei Auflösungen des ST, d.h. zwei Farbmodi und einen Monochrommodus. Folglich kann man sich den Kaufzweier Monitore sparen, da man quasi beide Geräte in einem besitzt. Man hat also mehr Platz auf dem Schreibtisch und dazu brillante Farben.

Schlichtes Grau

Der "kleine" Bruder des damals beschriebenen MultiSync ist der MultiSync GS, eines der jüngsten Produkte von NEC. Zu bewundern war er auf der Münchner Systems, damals aber noch nicht am ATARI ST. Bei ihm handelt es sich um einen Graustufen-Monitor, d.h., er wandelt Farben in verschiedene Graustufen um und verfügt somit über keine Farbdarstellung. Durch die Graustufen wird aber eine größere Trennschärfe als bei einer Farbdarstellung erreicht. ATARI ST-Besitzer können nun Spiele, Zeichnungen u.ä. ohne Probleme in Schwarzweiß bewundern, wie man an dem hübschen Bild aus dem CCD Spiel "KAISER" sehen kann (siehe Bild 2). Damit man die Qualität des Monitors besser beurteilen kann, haben wir für die mittlere und die hohe Auflösung je ein Beispiel aus 1st-Word fotografiert (siehe Bild 3 und 4).

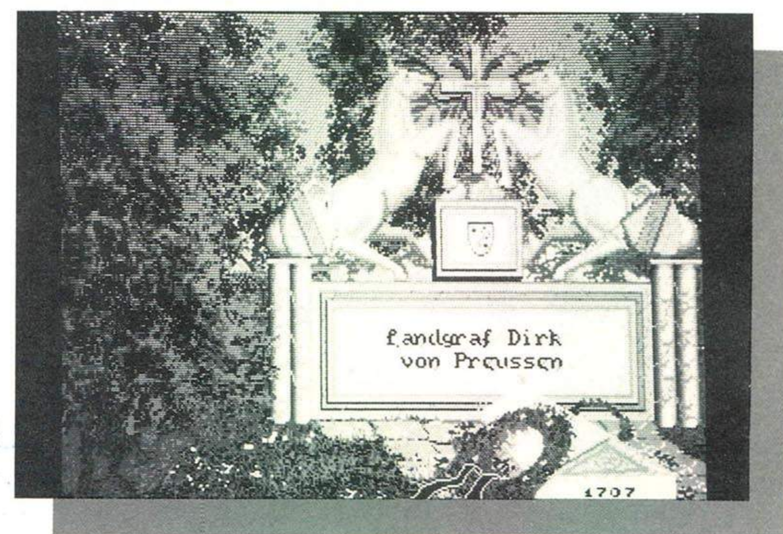


Bild 2: Umwandlung von Farbe in Graustufen am Beispiel von "KAISER"

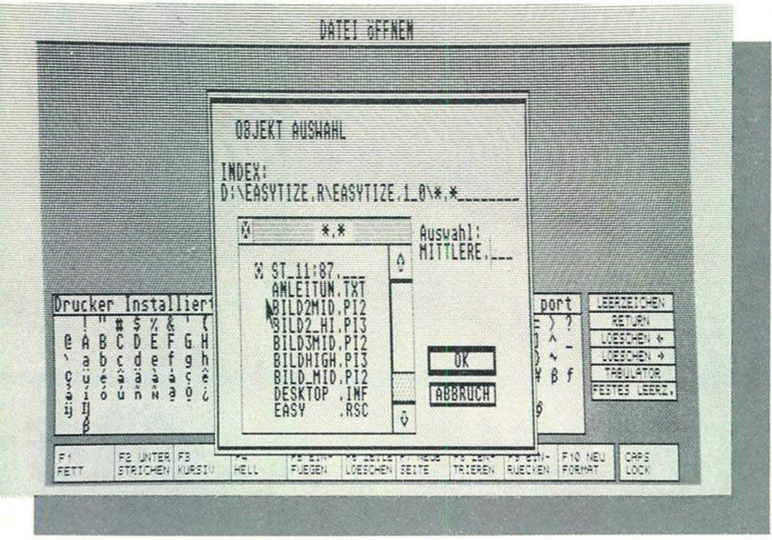
Der MultiSync GS verfügt über eine 14" (33,6 cm) entspiegelte Bildröhre und hat somit eine größere Nutzfläche als herkömmliche Monitore. Durch die Flachbauweise des Bildschirms finden keine Verzerrungen durch gewölbte Ränder statt. Man kennt dies ja schon von einigen Fernsehern der neueren Generation her. Zur serienmäßigen Ausstattung gehört ein Horizontal-/VertikalSchwenkfuß, mit dem man den Monitor leicht postieren kann. Der GS verfügt über eine automatische Erkennung der Synchronisationsfrequenz (bis zu 70 Hz). Normalerweise dient dies bei PC Kompatiblen zur Erkennung der jeweiligen Grafikkarte. Beim ST ist jedoch ein Umschalter zwischen monochromer und farbiger Darstellung notwendig (s.u.).

Im Test wurden durchweg gute Erfahrungen mit dem GS gemacht. Obwohl er nicht die Qualität des ATARI Monitors SM 124 (in Bezug auf die Schärfe des Bildes) erreicht, kann man doch in der hohen Auflösung gut mit ihm arbeiten. Die Farbumsetzung in verschiedene Graustufen erfolgte bis auf leichte Schwächen bei Rot gut.

Er ist für jeden, der gerne alle Auflösungen des ST nutzen und dafür nur wenig Geld ausgeben möchte, eine billige Alternative, denn der Preis beträgt nur ca. DM 650.–.



Bild 3: Darstellung in mittlerer Auflösung



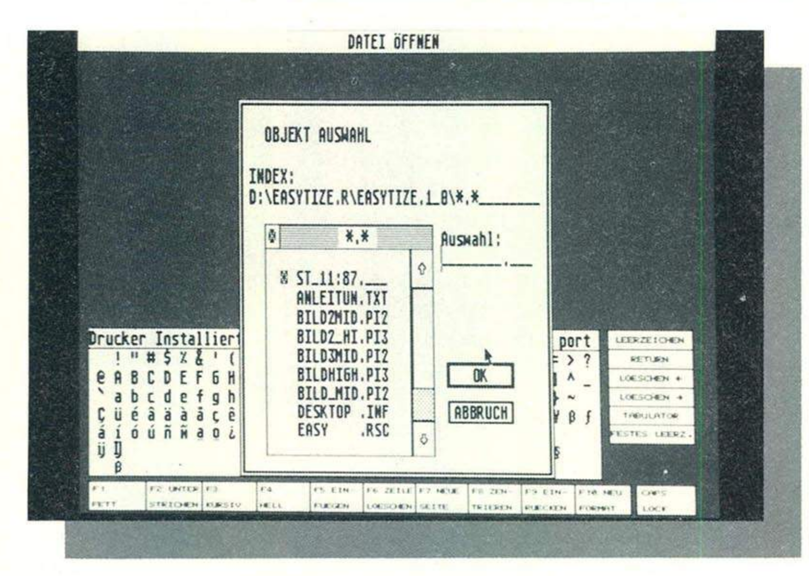
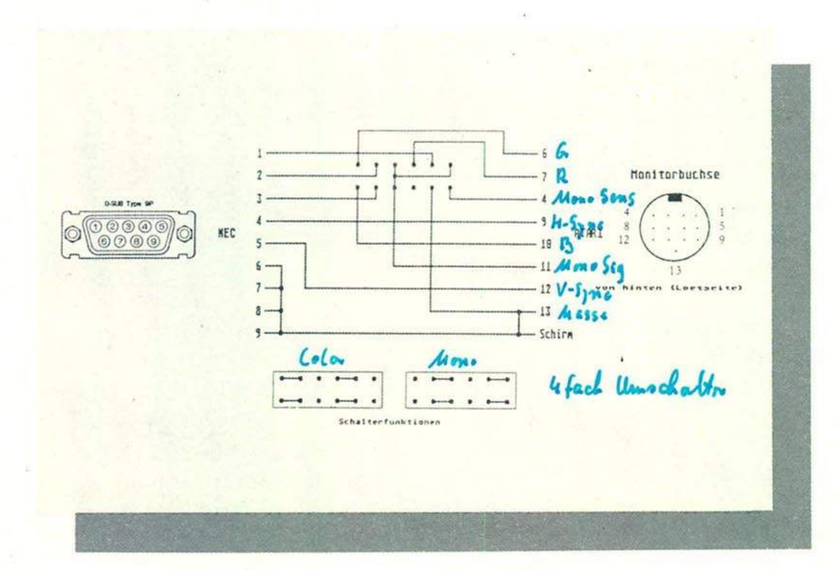


Bild 4: Darstellung in hoher Auflösung



Technische Daten

Bildröhre: 13.75" (35,6 cm) diagonal (schwarzweiß)
Auflösung: horizontal 900 x vertikal 700
Bandbreite: 30 MHz

Automatische Synchronisation:
Bildfrequenz: Bis 70 Hz
Zeilenfrequenz: Bis 31,5 kHz

Eingänge: TTL- und Analogsignal
TTL: Bis zu 64 Graustufen
Analog: Unbegrenzte Anzahl von Graustufen

Anschlußkabel: D-Sub 9 Pin

Anschluß gesucht

Genau wie beim Ur-MultiSync fehlt beim GS ein Tonteil, so daß die Bastler auf den Plan gerufen sind. Man muß einen kleinen NFVerstärker an Pin 1 der Monitorbuchse anschließen (Masse-Pin 13 nicht vergessen). Es war allerdings auch schon gerüchteweise von fertigen Kabeln mit eingebauten Tonteil zu hören.

Wer sich selbst ein Anschlußkabel für die Kommunikation zwischen ST und MultiSync GS basteln möchte (normalerweise wird der GS mit einem Kabel für PC-Kompatible mit einem 9-poligen Sub D-Stecker geliefert), dem haben wir das entsprechende Schaltbild abgedruckt. Benötigt wird dazu ein Vierfach-Umschalter, mit dem man zwischen hoher und niedriger bzw. mittlerer Auflösung wechseln kann. Es ist unbedingt darauf zu achten, abgeschirmtes 75 Ohm Kabel zu verwenden, da es sonst zu störenden Reflektionen auf dem Bidschirm kommen kann.

(HE)

OMEGA CAD – Der Grafik-Superstar



Wer jetzt denkt, es handelte sich um ein neues Grafikprogramm (noch eines !) darf sich freuen: denn Omega Cad ist kein banales Stück Software, sondern eine in massives braun-graues Blech gepackte Hardware-Erweiterung für den ST. Die Karte verwendet einen hochmodernen Grafikprozessor (Hitachi HD 63484), die Auflösung beträgt 1024*512 Punkte, von denen bis zu 820*512 gleichzeitig darstellbar sind.

Omega Cad treibt es ziemlich bunt – 256 Farben stehen gleichzeitig zur Verfügung, die dazu noch aus einer Palette von 256000 Farben ausgewählt werden können.

Die Grafikkarte hat einen eigenen Video-Speicher von 512 KByte Größe. Da der Grafikprozessor alle wesentlichen Grafikfunktionen selbständig und schnell ausführen kann, ist auch die Kommunikation zwischen Rechner und Grafikkarte problemlos.

Wer bisher immer neidvoll auf die bunten Bilder des Amiga geschaut hat, kann für ungefähr 3000.- DM eine Grafikerweiterung erstehen, die es in sich hat.

Für diesen Preis erhält man eine große Kiste, in der man nach einigem Wühlen in viel Verpackungsmaterial die Grafikerweiterung findet. Sie ist in ein ausnahmsweise nicht atarigraues, sondern braunes Blechgehäuse mit beiger Frontplatte, auf der sich nur ein dezenter Herstellerschriftzug und eine Betriebsanzeige finden, eingebaut. Das Ganze hat das Format 24*20*7 cm und ist sehr robust und sauber verarbeitet. Leider hat die Karte kein eigenes Netzteil; Atari 520 ST+Benutzer können sich also freuen, ein weiteres externes Netzteil harrt des Anschlusses. Stilvollerweise handelt es sich bei dem mitgelieferten Netzteil sogar um ein Original-Atari Gerät, das allerdings mit einem anderen Stecker auf der Sekundärseite ausgestattet wurde.

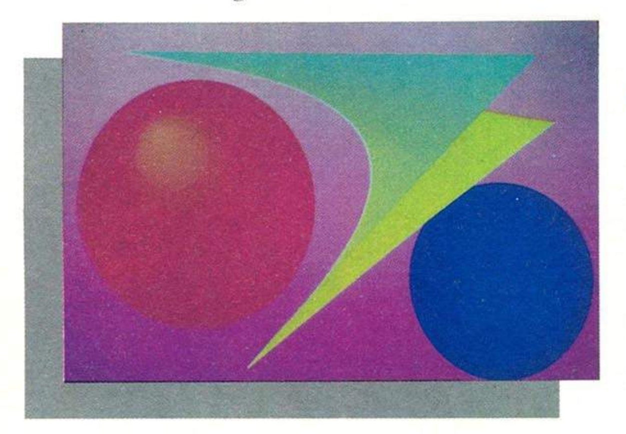
Ansonsten sind im Lieferumfang eine Bedienungsanleitung von knapp 90 Seiten und drei Disketten mit Programmierlibraries, Demoprogrammen und einem richtigen Malprogramm enthalten.

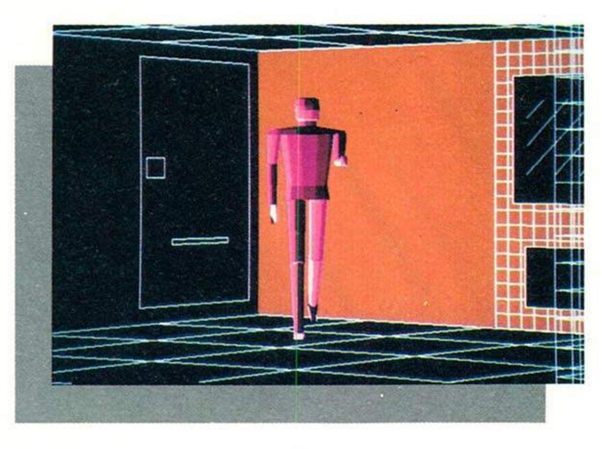
Die Erweiterung wird an den Romport angeschlossen. Dazu dient ein erfreulich langes Flachbandkabel, so daß man die Karte auch in die hinterste Schreibtischecke verbannen kann. Schließlich gibt es auch nur ein Bedienungselement, das man ab und zu betätigen muß, und das ist der An/Aus-Schalter auf der Rückseite des Gehäuses. Dort befinden sich auch die Anschlüsse für Netzteil und Monitor.

Apropos Monitor: Die Karte benötigt einen der modernen MultiscanMonitore, deren Synchronisierung sich selbstständig an das Eingangssignal anpaßt, weil eine sehr hohe Bildwechselrate verwendet und der sogenannte Interlace-Modus (eine Fernsehtechnik, bei der jedes Bild aus zwei Halbbildern zusammengesetzt wird, um eine höhere Pseudo-Auflösung zu erhalten) nicht benutzt wird.

Das Bild ist dadurch so richtig schön flimmerfrei. Wer einmal mit einem Amiga im Interlace-Modus gearbeitet hat, weiß, wovon ich spreche.

Wir haben die Karte mit einem NEC MultiSync und einem EIZO Flexscan ausprobiert. Beide Monitore erzeugen auch bei der höchsten Auflösung der Karte ein exzellentes Bild. Scharf und ruhig steht es da, an solche Bilder könnte ich mich gewöhnen...





Mit dem Romport-Anschluß gab es bei dem Rechner des Autors (ein uralter 520 ST+) übrigens Probleme. Laut Omega könnte dies an einer Änderung der Anschlußmaße des Atari liegen, auf neueren 520ern funktioniert es. Omega schickte zwar sofort zwei Ersatzstekker, aber auch mit diesen funktionierte die Karte nicht. Mit allen Atari 1040-Modellen, die der Autor in seiner Umgebung ergattern konnte, lief aber alles einwandfrei. Da andere Hersteller es auch geschafft haben, an jedem Rechner lauffähige Romport-Erweiterungen zu schaffen, wird eine Lösung dieses Problems sicher nicht lange auf sich warten lassen.

Die Karte wird, wie gesagt, an den Romport angeschlossen. Um sie zu benutzen, muß allerdings noch ein bißchen Treiber-Software geladen werden, die als Accessory vorliegt. Da dieses Accessory nichts tut, als nach dem Laden seine Existenz anzuzeigen und die erfolgreiche oder -lose Initialisierung der Grafikkarte anzuzeigen (es gibt da nichts zu bedienen), stellt sich natürlich die Frage, ob dafür ein Accessory-Platz verschwendet werden muß. Andererseits benutzen wohl sowieso die wenigsten überhaupt regelmäßig Accessories, so daß dieser Nachteil nicht besonders ins Gewicht fällt.

Wenn die Karte erfolgreich Anschluß gesucht hat und der Treiber geladen ist, was erst zu so richtig herzlichen Beziehungen zwischen Ihren Geräten führt, kann man damit beginnen, die mitgelieferten Programme auszuprobieren.

Da wäre zuerst mal das Malprogramm 'Assist'. Dieses stellt alle wichtigen Malfunktionen, die man bereits von anderen Malprogrammen für den ST kennt, zur Verfügung, aber eben mit einer viel höheren Auflösung und mit 256 Farben gleichzeitig. Diese 256 Farben kann man aus einer Palette von ca. 256000 Farben auswählen, so daß selbst so richtig fließende Farbverläufe kein Problem sind. Dafür gibt es Spezialfunktionen, sowohl für das Füllen mit Farbverläufen, als auch für die Berechnung einer Palette.

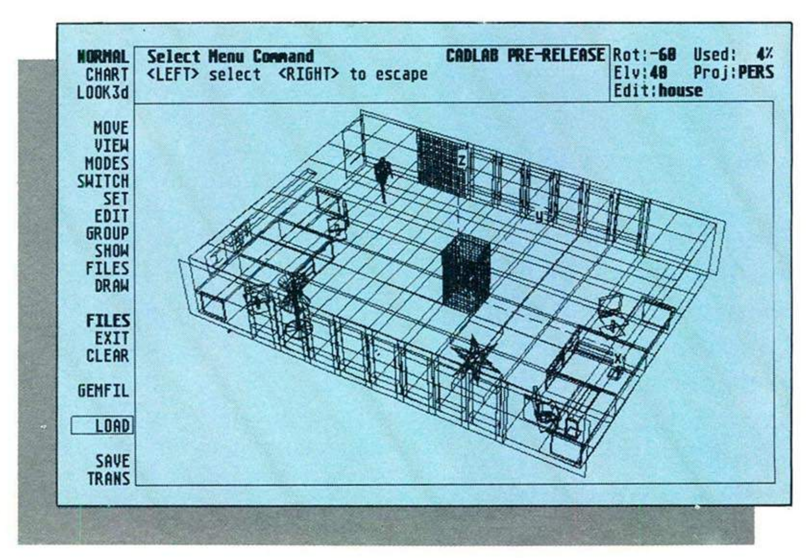
Durch die hohe Zeichengeschwindigkeit des auf der Karte verwendeten Grafikprozessors geht alles wunderbar schnell... Bis zu 4 Millionen Bildpunkte kann der Prozessor pro Sekunde darstellen, Flächenfüllen ist kein Problem. Das wirkt sich auch auf das Zeichnen von Kreisen und Rechtecken usw. aus; Keine Gummibandlinien werden gezeichnet, sondern immer gleich das richtige Objekt.

Praktischerweise verfügt die Grafikkarte auch über eine HardwareLupe. Das Vergrößern beliebiger Bildausschnitte nimmt also keine kostbare Zeit in Anspruch, sondern geschieht in Real-Time.

Selbstverständlich können Bilder geladen und gespeichert werden. Dazu wird natürlich ein eigenes Format verwendet, es ist aber auch möglich, Degas- und Neochrome-Bilder in das Spezialformat zu konvertieren.

Als besonderes Bonbon ermöglicht 'Assist' es, den Zeichenablauf, der zur Entstehung eines Bildes führt, zu speichern und wieder abzuspielen. Damit lassen sich sogar Animationen produzieren, wie man an einigen mitgelieferten Beispielbildern sehen kann.

Leider erfordert 'Assist' neben dem Multiscan-Monitor einen Schwarzweiß-Monitor. Dort wird dann das 'Assist'-Menü



dargestellt, während auf dem Farbmonitor die eigentliche Zeichnung entsteht, ungestört durch platzraubende Menüs.

Das Programm wird noch weiterentwickelt, z.B. zusätzliche Funktionen, die man schon von den besseren Farbmalprogrammen des ST kennt. Aber schon jetzt ist das Programm sehr brauchbar.

'Axiomega' ist ein 3D-Programm, das von einer holländischen Firma entwickelt wurde. Es ist eine Portierung von einem IBM-Programm, was man der Benutzeroberfläche leider ansieht. Dieses Programm benutzt für die Benutzerführung ebenfalls den normalen AtariMonitor, allerdings wahlweise auch in Farbe. Alle Bilder können aber auch in höherer Auflösung und Farbigkeit mit der Grafikkarte angezeigt werden.

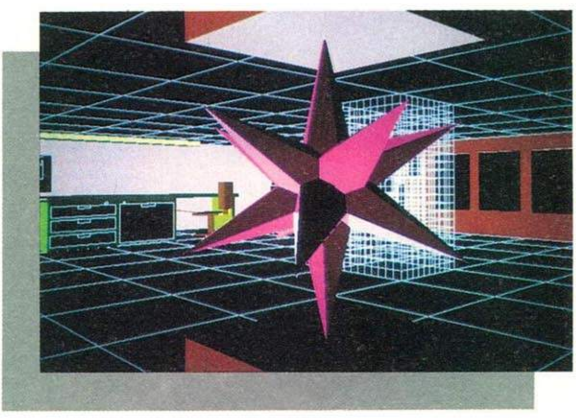
Bilder werden aus einzelnen Objekten zusammengesetzt. Man kann einzelne Punkte der Objekte (als Text) eingeben oder aber einige Grundformen wie Zylinder, Mauern oder Hausformen grafisch editieren. Die Seiten der Objekte können mit zweidimensionalen Details versehen werden. Leider läßt der Komfort der Eingabe stark zu wünschen übrig, was man aber hoffentlich der DemoVersion zurechnen darf.

Das ganze System soll u.a. Animationen erlauben, also Kamerafahrten durch die 'Welt'.

Bilder können sowohl als Drahtmodelle, als auch mit schattierten Oberflächen dargestellt werden. Dabei kann man die Lichtquellen beliebig positionieren. Die Schattierungsmethode ist einfaches Polygonshading, d.h. alle ebenen Flächen erhalten den gleichen Farbton. Zylinder, die ja aus sehr vielen Polygonen zusammengesetzt werden, sehen, der vielen möglichen Farbtöne wegen, aber schon richtig gut aus. Überhaupt ist der Unterschied zur Atari-Grafik bombastisch.

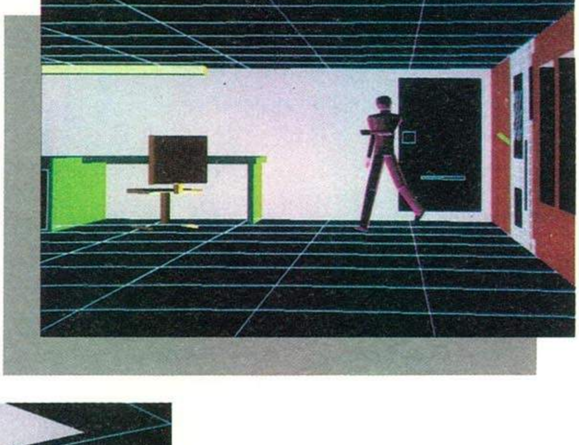
'Axiomega' kann zwar schon einiges, wird sich aber besonders hinsichtlich der Benutzerführung noch stark verbessern müssen.

Eine weitere Diskette enthält



Demobilder, die mit einem anderen 3D-Grafikprogramm erzeugt wurden, das ebenfalls bald erhältlich sein soll. Als Besonderheit ist dieses Programm in der Lage, mit Gouraud-Shading versehene Bilder zu errechnen. Gouraud-Shading ist ein Schattierungsverfahren, das weiche Farbverläufe auf einer Oberfläche erzeugt.

Am wichtigsten ist die dritte Diskette; sie enthält Bibliotheken für so ziemlich jede wichtige Programmiersprache. Im einzelnen: das DR-Entwicklungspaket (C), Megamax C, Lattice C, Omikron und GfA-Basic, CCD Pascal, DR AR68-Assembler, Makro-Assembler und



Modula II.

Über 50 Befehle, für so ziemlich jeden grafischen Zweck stehen zur Verfügung. Man kann Punkte setzen, Linien, Rechtecke, Kreise, Ellipsen zeichnen, Grafiktext verwenden. Sogar GEM-Fonts lassen sich benutzen. Apropos GEM: Mit einem Befehl kann man fast alle VDI-Befehle auf die Grafikkarte umlenken. Das ist für die Anpassung bestehender Software wichtig. Außerdem kann man die Farbpalette programmieren, ganze Speicherbereiche vom Atari zur Karte kopieren und umgekehrt, Bildschirmblöcke kopieren, die Bildschirmauflösung ändern und die Hardware-Lupe des Grafikprozessors programmieren. Auch die Maus kann praktischerweise auf den Omega-Bildschirm umgeschaltet werden. Schließlich verraten einige Befehle, daß noch Erweiterungen geplant sind: Ein Befehl namens 'digitize' soll der Übernahme von Video-Bil-

dern in den Computerspeicher dienen.

Das CAD-Programm CAD-Projekt wird gerade an die Grafikkarte angepaßt. Man kann nur hoffen, daß sich noch andere Hersteller an solche Anpassungen machen. Wenn man sich Tom Hudsons CAD 3D auf dieser Karte vorstellt – mit erweiterten Shading-Routinen..., das wäre hübsch.

Schon jetzt gibt es zwei Versionen der Karte: eine Version mit 256 Farben ohne Farbpalette und die von uns getestete Palettenversion mit einer 18-Bit Palette, also 256000 Farben. Eine größere Palette, die 24 Bit Tiefe besitzt (16,7 Millionen Farben), soll zum Erscheinungsdatum dieser Ausgabe bereits lieferbar sein. Bald wird es auch eine Version, die tatsächlich 16,7 Millionen Farben gleichzeitig auf den Bildschirm bringen kann, geben. Diese Karte soll auch über einen Echtzeit-Videodigitalisierer verfügen, sowie über eine Synchronisiereinrichtung, die es erlaubt, Videobilder im Bildschirmhintergrund laufen zu lassen. Aber dieses Modell wird wohl noch einige Zeit auf sich warten lassen.

Fazit:

Wer mit der Atari-Grafik nicht zufrieden ist und seinen Rechner für wirklich hochwertige Grafikaufgaben verwenden will, sollte einen Blick auf die Omega-Grafikkarten werfen. Die Bildqualität ist ausgezeichnet, ebenso die Verarbeitung. Man kann nur hoffen, daß einiges von der existierenden Atari-Software für die Karte angepaßt wird. Sonst ist sie vor allem auf Selbstprogrammierer interessant, die mit den wirklich ausführlichen Bindings wohl keinerlei Probleme haben werden. Auch das Preis/Leistungsverhältnis ist in Ordnung.

'Normalbenutzer',

die nicht selbst programmieren wollen, sollten zuerst auf die verfügbare Software achten. Ansonsten: empfehlenswert.

CS

Benelux: Cat + Korsch, Rotterdam 010-4 50 76 96

P 2200 – DAS PREIS-LEISTUNGS-GENIE

Die Computer-Anwender haben Grund zum Jubeln!

Genial – endlich ein Drucker, der für Einsteiger, Aufsteiger und Semiprofis geeignet und vor allem erschwinglich ist. Denn NEC erschließt Ihnen jetzt die modernste 24-Nadel-Drucktechnologie zu einem erstaunlich günstigen Preis.



Vielseitige, durchdachte Papierzuführungen.

NEC Pinwriter P 2200 mit automatischer Einzelblattführung

Erleben Sie eine neue Dimension: gestochen scharfen Korrespondenzdruck mit ungewöhnlich reicher Schriftartenauswahl, brillante Grafik-Darstellung, bequeme Druckersteuerung und integrierte Papierzuführungen.

Warum also tief in die Taschen greifen, wenn es schon für wenig Geld 24-Nadel-Technologie mit allen Raffinessen gibt?

PROFIQUALITÄT ZUM AMATEURPREIS

NEC ist mit seinen 24-Nadel-Druckern in Deutschland marktführend.

Was den P 2200 als echten Profi auszeichnet, sind seine hohe Auflösung von 360 x 360 dpi, ein halbes Dutzend serienmäßiger Schriftarten und eine Reihe praktischer Papierzuführungen. Zum Beispiel können Sie zwischendurch einen Brief drucken, ohne daß das Endlospapier extra herausgenommen werden muß.

Durch seine volle Kompatibilität mit den NEC Pinwritern der 24-Nadel-Serie harmoniert der P 2200 mit allen wichtigen Software-Paketen.

EIN NEC DRUCKER FÜR JEDERMANN

Endlich braucht niemand mehr auf die bewährte NEC Produkt- und Druckqualität zu verzichten.

Hervorragende Druckqualität durch bewährte 24-Nadel-Technologie.

Eine breitgefächerte Gruppe – vom Schüler über den Heimanwender bis hin zum Freiberufler – findet im P 2200 die ideale Drucklösung. Anwendern, die schon seit langem auf der Suche nach einem preisgünstigen Drucker für ihren Computer sind, eröffnet er die Möglichkeit, Druckergebnisse in bewährter NEC-Qualität zu erzielen.



12 Schriftartenkassetten zusätzlich erhältlich.

Damit ist der P 2200 die optimale wirtschaftliche und technische Alternative für alle, die sich bei gleichem finanziellen Aufwand bisher nur mit antiquierten 9-Nadel-Druckern begnügen mußten.

Weitere Informationen zum P 2200 erhalten Sie von Ihrem NEC Drucker-Fachhändler.

NEC

NEC Deutschland GmbH

Klausenburger Straße 4, 8000 München 80
Tel.: 0 89/9 30 06-0, Telefax: 0 89/93 77 76/8
Telex: 5 218 073 und 5 218 074 nec m

Eine kleine Box mit viel Inhalt

– Die Fileselektorbox –

Da wir sehr viele Anfänger unter unseren Lesern haben, möchten wir unter anderem mit diesem Artikel dazu beitragen, den Umgang mit dem Rechner zu erleichtern. Doch auch denjenigen, die schon länger mit ihrem ST leben, ist vielleicht noch nicht alles über die kleinen Kniffe des Betriebssystems bekannt. Viele Handgriffe sind nötig, um den ATARI ST komfortabel benutzen zu können. Ein sehr oft auftauchendes Objekt im Umgang mit diesem Rechner ist die DATEIAUSWAHLBOX, die hier näher beleuchtet werden soll, um einen einfacheren Einstieg in die Welt des ST's zu ermöglichen...

Sitzt man zum ersten Mal vor dem Atari ST, so hält man sicherlich das Einführungsheft, das ATARI mitliefert, in der Hand, und läßt sich in die Geheimnisse des DESKTOPS einweihen. Startet man dann das erste Programm, bei vielen wird es zum Beispiel eine Textverarbeitung sein, in der als erstes eine Datei ausgewählt werden muß, so verlassen einen die liebgewonnenen Fenster, die es ermöglichen auf dem Desktop Programme und Dateien anzuwählen. Anstatt dessen findet man sich fast immer vis-à-vis einer Box, die zwar einem Desktopfenster ähnelt, aber doch von ihm verschieden ist. Diese Box (groß und breit auf diesen Seiten zu finden) nennt sich auf neudeutsch FILESELEKTORBOX oder auch DATEIAUSWAHLBOX. Mit jener Box wollen wir uns nun beschäftigen.

Die Fileselektorbox besteht aus mehreren Teilen: Zum einen die oberste editierbare Zeile, die Index-Zeile, über der das Wort "INDEX" steht. Hierbei bedeutet Editieren, das Einfügen von Buchstaben über die Tastatur durch den Benutzer. Zum anderen findet sich eine kleinere editierbare Zeile, über der 'AUSWAHL:' steht. Desweiteren befindet sich in der Fileselektorbox eine kleine Box, in der ein Ausschnitt des angewählten Disketteninhaltes zu finden ist mit den zugehörigen Pfeilen, der WILDCARD, der CLOSEBOX und dem SCHIEBER (diese drei Begriffe werden unten näher erläutert). Außerdem enthält sie zwei anwählbare Tasten, in denen 'OK' und 'ABBRUCH' stehet.

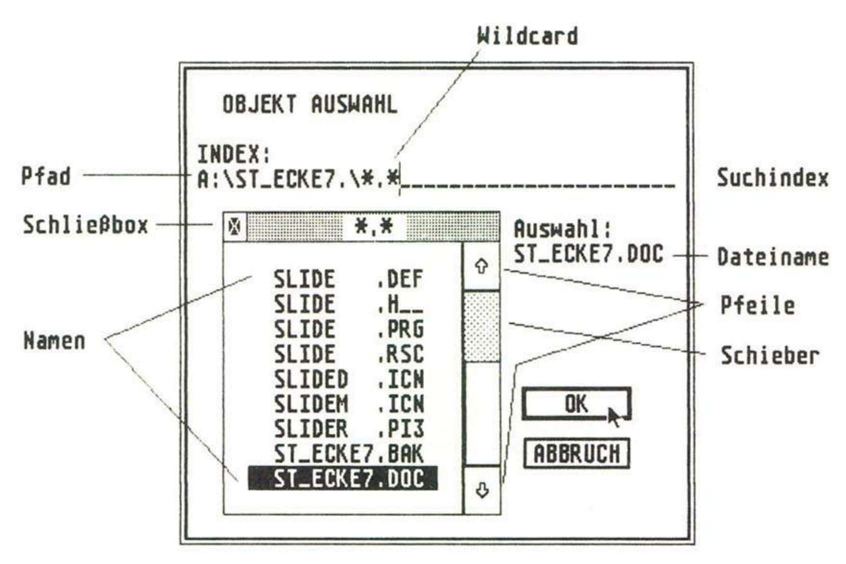
Zunächst ein Wort zum Eingeben von Buchstaben in die beiden editierbaren Felder:

Erscheint die Fileselektorbox auf dem Bildschirm, so befindet sich ein großer, dünner, senkrechter Strich im rechten Edit-Feld, der Cursor. Wenn Sie nun ein Zeichen eingeben, so erscheint der Buchstabe in diesem Feld und der Cursor rückt eins nach rechts. Wollen Sie in das obere Edit-Feld, so drücken Sie die Cursor-Hoch-Taste der Cursor befindet sich dann im oberen großen Edit-Feld. Zum kleinen Feld gelangen Sie dann wieder mit der Cursor-Runter-Taste. Wollen Sie das ganze Feld löschen, so brauchen Sie dazu nicht jeden Buchstaben mit BACKSPACE (rechts oben) zu löschen, sondern es reicht, wenn Sie die ESC-Taste (links oben) drücken und der gesamte Text in der Zeile verschwindet.

Das INDEX-Feld:

In diesem Feld steht der sogenannte Pfadname einer Datei. Nun, was ist ein PFAD? Wie ein normaler Pfad gibt er den Weg zu einem Ziel an, das in unserem Fall die Datei ist. Wie Sie sicherlich wissen, kann man in solch modernen Rechnern wie dem ST die Diskette in Ordner unterteilen. Nun kann allerdings in einem Ordner wieder ein Ordner sein. (Zur Verdeutlichung schauen Sie sich bitte Bild 1 an). Diese Aufteilung von Disketten sollte man sich von Anfang an angewöhnen, auch wenn ein Ordner auf der Diskette ein Kilobyte verbraucht. Ein Gesamtpfad besteht also aus der Diskettenstationskennung (A,B,C,D etc.), der Ordnerverschachtelung und dem anschließenden Namen. Der Name besteht übrigens wieder aus zwei Teilen, dem eigentlichen Namen, der acht Buchstaben sein darf, und dem Extender (Erweiterung), der eine Länge von bis zu drei

Buchstaben hat. Diesen Extender sollte man der Übersicht wegen nur zum Bezeichnen des Dateityps benutzen. Dabei haben sich schon einige Bezeichnungen ergeben, die häufig benutzt werden : .C (Quellcode der Programmiersprache C), .PAS (PASCAL), .BAS (BASIC), .DOC (englischer Text), .DOK (deutscher TEXT), .PRG (Programm, das unter GEM läuft) usw..



1
Die Fileselektorbox

WILDCARDS

Anstatt des Namens steht nun im Indexfeld an letzter Stelle eine sogenannte WILDCARD oder ein Joker. Dies ist ein Platzhalter für eine Anzahl von Dateien und dient dazu eine Auswahl von Dateien anzuzeigen.

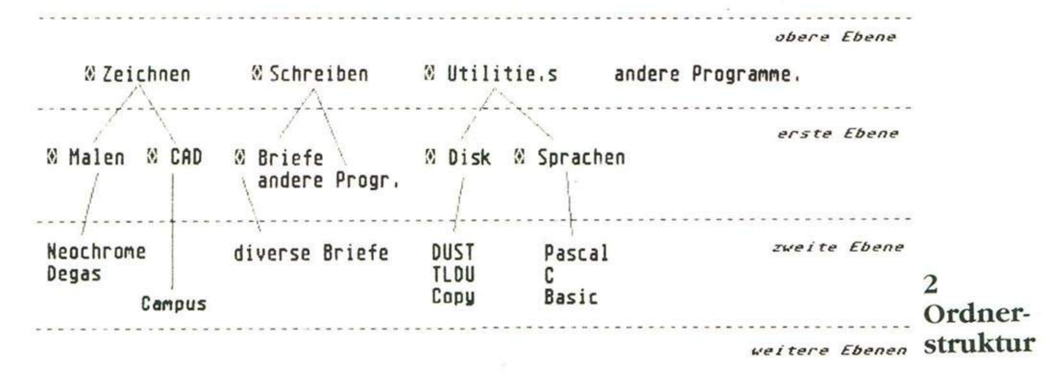
Dazu werden die Zeichen '*' und '?' benutzt. Das Fragezeichen steht als Platzhalter für ein Zeichen, während das Asterisk ('*') eine ganze Zeichenkette ersetzt. Zur Verdeutlichung schauen wir uns ein paar Beispiele an:

D?CK.PRG :
zeigt alle Dateien an, die die angegebenen Buchstaben enthalten, wobei der zweite Buchstabe beliebig ist. Zum Beispiel DICK.PRG, DUCK.PRG, DRCK.PRG usw.

BU*.DOC :
Es werden alle Dateien aufgelistet, die mit BU beginnen und den Extender DOC besitzen. Zum Beispiel BUCH.DOC, BULK.DOC ,BUECHER.DOC etc.

BU*.* :
Es werden alle Dateien aufgelistet, die mit BU beginnen. Der Extender wird hier außer acht gelassen, da auch in ihm eine Wildcard vorhanden ist.

***.DO? :**
Alle Dateien (der eigentliche Name ist wegen des Asteriks unwichtig) die im Extender ein DO stehen haben, werden angezeigt. Beispiel: Alle Textdateien englisch (DOC) wie auch deutsche (DOK) sind zu finden.



2
Ordnerstruktur

Um ein wenig mit dieser Dialogbox herumzuspielen, haben wir für C, PASCAL und BASIC kleine Programme beigelegt, um eine Dateiauswahlbox auf den Bildschirm zu bringen. Bitte geben Sie eins dieser Programme ein und starten es Sie können auch ein beliebiges Programm zur Hand nehmen, von dem Sie wissen, daß diese Dateiauswahlbox in ihm vorhanden ist.

Der Umgang mit der Fileselektorbox

Wenn Sie nun die Box auf dem Bildschirm haben, legen Sie in das Laufwerk eine Diskette ein. Am besten zum jetzigen Herumspielen ist eine Diskette mit ein paar Ordnern und vielen Dateien geeignet. Bei den abgedruckten Programmen befindet sich im Indexfeld folgender Eintrag: 'A:*.*', wobei sich der SuchErsatz des Namens im grau unterlegten Feld über den Dateinamen wiederfindet. 'A:*.*' bedeutet, daß alle Dateien eingeblendet werden, die sich in Laufwerk A befinden. Das Feld unter AUSWAHL ist leer. Bitte klicken Sie nun eines der angezeigten Felder an. Dieser Dateiname wird in das Auswahlfeld übernommen. Sie brauchen ihn also nicht unbedingt mit der Tastatur einzugeben, was auch in der Praxis selten geschieht. Bewegen Sie bitte jetzt den Cursor in das obere Feld und tauschen Sie den Such-Index '*.*' mit einem Extender auf ihrer Diskette vorkommt, zum Beispiel '*.PRG'. Noch werden die Dateien nicht wie gewünscht angezeigt. Sie müssen erst mit der Maus in das Feld klicken, in dem sich die Dateinamen befinden. Nun sehen sie nur noch die Dateien, die diesem Such-Index entsprechen, wobei es natürlich sein kann, daß keine Dateien mehr angezeigt werden. Wollen Sie wieder den Such-Index '*.*', so klicken Sie bitte die graue Leiste über den Dateinamen an, in der auch der Suchindex vorhanden ist. Dies können Sie sich zunutze machen, indem Sie zum Beispiel 'B:' ,'C:' etc. im Index-Feld eingeben und dann das grau unterlegte Feld anklicken, denn dann wird automatisch '*.*' an-

gehängt. Möchten Sie aber den Such-Index 'A:*.*', so geht dies sehr einfach: löschen Sie das Indexfeld, in dem Sie den Cursor in ihm positionieren und die Taste ESC drücken (siehe oben). Jetzt klicken Sie die grau unterlegte Such-Index-Leiste an und das Such-Index-Feld wird auto- matisch mit 'A:*.*' aufgefüllt!

Achtung: Wenn Sie die Diskette wechseln und den Index nicht geändert haben, was der Normalfall ist, so müssen Sie zum Laden des neuen Disketteninhaltes das graue Feld anklicken. Sie brauchen keine Angst zu haben, daß dann Ihr Index (z.B. *.PAS) verlorengeht, er bleibt erhalten und wird nicht in '*.*' umgewandelt.

Wollen Sie einen Ordner öffnen, so klicken Sie diesen einfach an, dann finden sich im Dateinamenfeld nur noch die Dateien des Ordners, der Such-Index ist erhalten geblieben, und im Pfad ist der Name des Ordners eingefügt worden. Wollen Sie diesen Ordner wieder schließen, wählen Sie die CLOSEBOX (Schließbox siehe großes Bild) an, und Sie befinden sich wieder ein Stufe über diesem Ordner und der Ordnername ist aus dem Pfad entfernt worden.

Sicherlich ist es einleuchtend, daß bei einer großen Anzahl von Dateien nicht alle in diesem kleinen Fenster Platz haben können, so daß Sie nur immer einen (kleinen) Ausschnitt sehen. Um diese Bereiche umzuschalten und um weiterzublättern, sind die Pfeile und der Schieber vorgesehen. Wenn Sie nun eine Diskette haben, auf der mehr als neun Dateien sind, erkennen Sie zwischen den beiden Pfeilen eine Box, die grau unterlegt und zum Teil weiß ist. Nun müssen Sie sich vorstellen, daß der Bereich zwischen den beiden Pfeilen den gesamten Bereich der Diskette symbolisisiert, wobei Sie aber nur den Teil sehen, der weiß ist. Am besten kann man sich dies verdeutlichen, wenn man sich vorstellt, daß der weiße Bereich ein Fenster ist, das man über dem Disketteninhalt (in der Fachsprache auch Directory genannt) schiebt. Sehen Sie nur einen weißen und keinen grauen Teil, so bedeutet das, daß der gesamte Disketteninhalt angezeigt wird, sehen Sie nur eine sehr kleine Box, so sind die angezeigten Dateinamen nur ein kleiner Ausschnitt aus dem Gesamtinhalt der Diskette. Wollen Sie nur eine Datei weiter nach unten blättern, so betätigen Sie bitte den nach unten zeigenden Pfeil, der andere Pfeil blättert entsprechend nach oben. Bei einer großen Anzahl von Dateien wäre das aber zu langwierig, weshalb es eine schnellere Methode gibt, vorwärtszukommen: Klicken Sie den weißen Bereich zwischen den beiden Pfeilen an,

```
 1:
 2:  Do
 3:    Fileselect "c:\*.bas","beispiel.bas",A$
 4:    If A$="" Then
 5:      Alert 1,"ABBRUCH gedrückt !!!",1,"weiter|Ende",A
 6:    Else
 7:      Print A$
 8:    Endif
 9:    Exit If A=2
10:  Loop
```

Listing 1: In GFA Basic

```
 1:
 2:
 3:  #include <gemdefs.h>
 4:
 5:  char fpath[64], fname[64];
 6:  int  key;
 7:
 8:  main()
 9:  {
10:     appl_init();
11:
12:     strcpy(fpath,"A:\*.*");
13:     strcpy(fname,"BEISPIEL.PRG");
14:     fsel_input(fpath,fname,&key);
15:     if(key == 1)                      /* OK-Taste gedrückt */
16:     {
17:        printf("Ok gedrückt: Index: %s\n              Name:
           %s\n",fpath,fname);
18:     }
19:     else                              /* Abbruch gedrückt */
20:     {
21:        printf("Abbruch gedrückt: Index: %s\n
           Name: %s\n",fpath,fname);
22:     }
23:     gemdos(7);                     /* Auf Tastendruck warten */
24:     appl_exit();
25:  }
```

Listing 2: In C

```
10 MOUSEON
20 Pfad$="A:\*.*"
30 FILESELECT (Pfad$,Datei_Name$,Ok)
40 IF Ok
50  THEN PRINT "Datei angewählt"
60  ELSE PRINT "Abbruch gedrückt"
70 ENDIF
```

Listing 3: In Omikron Basic

```
 1:  program file_select;
 2:
 3:  const {$i gemconst.pas}
 4:  type  {$i gemtype.pas}
 5:
 6:  VAR pfad, name: string;
 7:
 8:  {$i gemsubs.pas}
 9:
10:  Begin
11:    if init_gem>=0 then begin
12:       pfad:='A:\*.*';
13:       name:='';
14:       if get_in_file(Pfad,Name) then
15:            writeln ('OK gedrückt');
16:       else
17:            writeln ('Abbruch gedrückt');
18:       exit_gem;
19:    end;
20:  end.
```

Listing 4: In Pascal

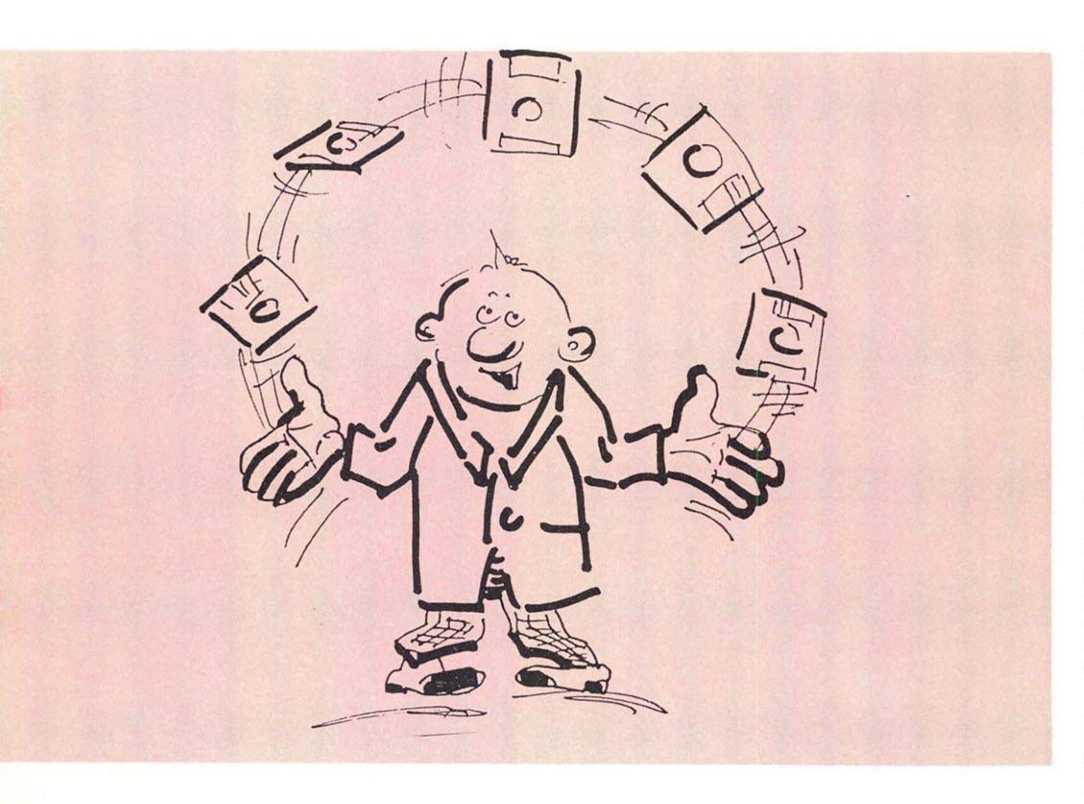
und lassen Sie die Maustaste gedrückt. Wenn Sie nun die Maus nach oben oder unten schieben, verschieben Sie den weißen Bereich (oder auch Ausschnitt) mit. Schieben Sie ihn dann an die gewünschte Stelle und lassen Sie die Maustaste los Sie sehen jetzt den entsprechenden Ausschnitt des Disketteninhaltes.

Soweit, so gut: Sie wissen, wie man den entsprechenden Directory-Ausschnitt in der Box zu sehen bekommt. Wie oben erwähnt, wählt man den Namen, den man möchte, dadurch aus, daß man ihn selektiert und daraufhin erscheint er dann im Dateinamenfeld. Danach muß man noch die Taste OK anklicken, wenn man diese Datei wirklich anwählen wollte. Sollte man aber zum Beispiel irrtümlich diese Fileselektorbox angewählt haben, so drücken Sie ABBRUCH. Das Programm, in dem Sie sich befinden, wertet dann diese Ausgangstaste entsprechend aus. Wenn Sie die Taste RETURN drücken, entspricht das übrigens dem Anwählen der Taste OK !! Wem das Selektieren des Namens und dann Anklicken der Taste OK noch zu viel Aufwand bedeutet und zu langsam ist, dem kann geholfen werden. Es geht noch schneller: Den gleichen Effekt erzielen Sie, in dem Sie über dem gewünschten Dateinamen zweimal schnell klicken (Doppelklick).

Wir hoffen Ihnen mit dieser Einführung ein wenig über ihre Arbeitsschwierigkeiten hinweggeholfen oder vielleicht ein paar Tips gegeben zu haben. Vielleicht klingt das obige etwas verwirrend, aber wenn Sie es einmal ausprobieren, werden Sie es sicherlich auch bald so gut beherrschen, wie manche Computerfreaks, die in hoher Geschwindigkeit mit der Maus über den Bildschirm huschen...

(SH)

FLOPPY-PIELEREIEN



Teil VI: HEFTIGE NACHWEHEN

Eigentlich wollte ich ja nach dem fünften Teil dieses Kurses (im Novemberheft) in der Versenkung verschwinden. Doch Sie haben zu früh aufgeatmet: Der Floppykurs erfährt eine – wenn auch kurze – Fortsetzung. Häufig gestellte Leserfragen, die alle interessieren könnten, sollen einmal beantwortet werden, außerdem möchte ich noch zu dem einen oder anderen Thema einen Nachtrag bringen.

Frage 1: Kann ich die Anzahl der FATs im Bootsektor auch auf 1 abändern und so mehr Platz auf der Diskette schaffen?

In der ST-Computer 7/87 hatte ich noch spekuliert, daß dies wohl möglich sei. Mittlerweile haben Tests aber ergeben, daß GEMDOS auf andere FAT-Anzahlen als zwei nicht vorbereitet ist. Es ist und bleibt eben der Schwachpunkt im Betriebssystem des ST, das GEMDOS. An anderer Stelle in diesem Heft finden Sie im GEMDOS-Kurs von Alex Esser (sei gegrüßt!) mehr Informationen zu diesem leidigen Thema.

Frage 2: Welche Manipulationen sind mit den fiktiven Verzeichniseinträgen '.' und '..' möglich?

Wahrscheinlich keine. Im Teil 2 des Floppykurses habe ich zwar erwähnt, daß diese beiden Einträge in jedem Unterverzeichnis automatisch gebildet werden und durch eine Verzeigerung die Orientierung im Ordnergestrüpp erleichtern können; doch hat sich auch hier herausgestellt, daß GEMDOS diese Einträge überhaupt nicht benutzt, um sich zurechtzufinden. '.' und '..' sind nur beibehalten worden, um mit MSDOS-Disketten kompatibel zu bleiben (Hallo Alex!).

Frage 3: Welche Probleme gibt es beim Einsatz von HYPERFORMAT?

Nicht mehr viele. Ich hatte ja vor längerer Zeit gemutmaßt, daß eventuell verschiedene Controllerserien an recht merkwürdigen Toleranzen schuld sein könnten, die HYPERFORMAT zu schaffen machten. Mittlerweile ist HYPERFORMAT aber so sicher geworden, daß diese minimalen Schwankungen wohl nicht mehr den Ausschlag geben.

Weiterhin wichtig bleibt aber die Drehzahl Ihres Laufwerks. Sie muß im Bereich von 300 bis 302 Umdrehungen pro Minute liegen und nicht, wie früher erwähnt, im Bereich bis zu 304 Umdrehungen. Auf diese Zahl war ich verfallen, weil mir das Public-Domain-Programm SPEED.TOS immer 1.5 Umdrehungen zuviel anzeigte; das heißt, ein Laufwerk, das die Diskette 300mal pro Minute komplett herumwirbelt, hat laut SPEED.TOS 301.5 Umdrehungen. Mittlerweile habe ich mir meine eigenen Meßprogramme geschrieben, die ihren Dienst korrekt verrichten. Sollten Sie selbst SPEED.TOS verwenden, können Sie bei Drehzahlanzeigen bis 304 Um-

drehungen noch ruhig bleiben; darüberhinaus wird es aber langsam brenzlig für HYPERFORMAT. Warum?

HYPERFORMAT benötigt in den jetzigen Versionen etwa 6250 Bytes auf einer Spur. Bei einer normalen Drehzahl hat man die auch, sogar etwa 50 bis 70 Bytes mehr (wenn auch der Hersteller des Floppycontrollers das nicht wahrhaben will). Dreht das Laufwerk aber zu schnell, wird der Platz enger; es kann dann passieren, daß der letzte Sektor auf der Spur nicht mehr komplett geschrieben werden kann.

Es kommt auch vor, daß sich bei zu fixen Laufwerken die Datenfelder der Sektoren etwas verbreitern und die nachfolgenden Lückenbytes überschreiben; geschieht das mehrmals, werden die Synchronisationsbytes des nachfolgenden Adreßfeldes angeknabbert, und die Katastrophe ist nicht mehr weit (Zähne fest zusammenbeißen, Hände in den Nakken, Atem anhalten und beten).

Um dies alles brauchen Sie sich aber meist keine Gedanken zu machen, weil die Laufwerke im Normalfall schon ab Werk richtig justiert sind. Nun ist aber bei ATARI eigentlich nichts normal, und deswegen gibt es immer wieder Laufwerke, die nicht genau genug eingestellt sind, was man aber selbst leicht nachholen kann. Bei der alten SF314 (Epson-Laufwerk, zu erkennen am lauten Röhren des Drehmotors und der kleinen Auswurftaste) geht man zum Beispiel folgendermaßen vor:

1. Laufwerk aufschrauben (vorher alle Kabel weg)
2. Anschlußplatine abziehen, Metallumhüllung des eigentlichen Laufwerks abschrauben.
3. Laufwerk umdrehen; auf der Unterseite findet sich ein kleiner Trimmer mit der Bezeichnung VR01, der für die Geschwindigkeitsregelung zuständig ist.
4. Anschlußplatine wieder ans Laufwerk stecken, Kabel installieren, Rechner und Laufwerk anschalten (keine Angst, es laufen nur 5V- und 12V-Leitungen durchs Laufwerk, das beeindruckt Ihren Herzschrittmacher nicht im geringsten), Meßprogramm für die Geschwindigkeit laden (siehe auch Listing 1 und die dazugehörige Dokumentation) und starten.
5. Geschwindigkeit auf 300 Umdrehungen pro Minute einstellen (vorsichtig an VR01 drehen!).
6. Alles wieder ausschalten und zusammenschrauben, leise vergnügt vor sich hin pfeifen, Rechner anschalten und alle Disketten auf HYPERFORMAT umformatieren.

Im Listing 1 und der zugehörigen Dokumentation finden Sie eine Routine, die Sie in den *Extended Minimon* (siehe ST-Computer 9/87 und 10/87) einbinden können. Sie mißt die aktuelle Drehzahl und errechnet auch einen Durchschnittswert.

Bei anderen Laufwerkstypen (selbst von den Original-ATARI-Laufwerken gibt es mindestens vier Versionen, von den Laufwerken im 1040 oder in den MegaSTs ganz zu schweigen) mag der Regler anders heißen und auch anders zu erreichen sein, das Prinzip bleibt dasselbe. Eine ausführliche Anleitung für einige ausgewählte Laufwerkstypen finden Sie in meinem Floppybuch, das demnächst wohlfeil im Handel zu erstehen ist. Dort finden Sie übrigens nebst anderen Überraschungen auch eine neue Über-Giga-Version von HYPERFORMAT (V3.0).

Frage 4: Mein Laufwerk schlägt schon bei Spur 80 an; kann man das irgendwie ändern?

Man kann eventuell. Zu diesem Thema hat mich sogar einmal ein Leser angerufen, der sich etwas ausgedacht hatte. Er hatte irgendwelche mechanischen Begrenzungen in seinem Laufwerk einfach abgeschleift, was natürlich ein wenig hemdsärmlig ist. Aber es funktionierte dann doch so gut, daß er – Zitat – "den Lesekopf so weit nach innen bewegen kann, bis er aus der Halterung fällt". Auf diese Weise kann man einige innere Spuren (80 bis 85) noch zur Datenspeicherung nutzen und – per HYPERFORMAT – über 960000 Bytes auf eine Diskette quetschen. Ich möchte Ihnen davon abraten, höhere Spuren als Spur 85 zu benutzen, weil dort Ihre Daten wirklich nicht mehr sonderlich sicher wären. Leider habe ich nichts mehr von diesem mutigen Zeitgenossen gehört; er möge sich doch, falls er diese Zeilen hier liest, noch einmal bei mir melden. Danke.

MEA MAXIMA CULPA

So, und nun möchte ich mich für einige meiner Fehlgriffe im Floppykurs entschuldigen. Ein paar Details sind doch nicht so korrekt gewesen, wie ich sie nach bestem Wissen und Gewissen ausgeforscht und niedergeschrieben hatte. Zum Glück handelt es sich vergleichsweise um Kleinigkeiten.

In der ST-Computer 11/87 verhinderte beispielsweise ein dummer Tippfehler, daß Sie die wahre Codierung der Steprate in Controllerbefehlen erfuhren. Hier die korrigierte Tabelle (Tabelle 1):

Stepratenfeld		Steprate
r1	r0	
0	0	6ms
0	1	12ms
1	0	2ms
1	1	3ms

(Tabelle 1: Das Stepratenfeld in FDC-Befehlen, diesmal hoffentlich richtig abgetippt)

Im dritten Teil habe ich die Belegung der Adresse $FF8606 (DMAModus-Register) erläutert. Dieses Register ergibt – wenn man es liest – den aktuellen DMA-Status. In Bit 0 steht dann aber nicht, wie irrtümlich abgedruckt, eine 0, wenn kein DMA-Fehler aufgetreten ist, sondern eine 1.

Mein schlimmster Schnitzer aber – und das haben auch einige erkannt und mich je nach Temperament mehr oder weniger verärgert darauf hinge-

wiesen – war die Darstellung der FAT-Struktur im ersten Teil des Floppykurses. Dort hatte ich folgenden Anfang einer FAT angegeben:

F7 FF FF 03 40 00 05 FF F0

Eine solche FAT kann unter TOS gar nicht existieren, die letzten beiden Bytes müßten eigentlich vertauscht sein. Schon damals hatte ich ja einen Hilferuf an Sie, verehrte Leserin, lieber Leser, abgesetzt, da ich selbst anfangs mit der FAT so meine Schwierigkeiten hatte. In der Zwischenzeit habe ich durch eigene Experimente und nicht zuletzt durch einige nette Zuschriften viel darüber gelernt und möchte das an dieser Stelle weitergeben.

FATsucht, Teil II

Auf der FAT ist die Belegung einer Diskette mit Dateien vermerkt. Dabei werden aber immer komplette Cluster belegt; das sind normalerweise zwei zusammenhängende Sektoren. Eine FAT könnte nun so anfangen:

F7 FF FF 03 40 00 05 F0 FF

Jedem Cluster werden in der FAT 12 Bit (!) zugewiesen, das sind drei *Hexadezimalziffern*. TOS und Brüderchen MS-DOS beginnen die Nummerierung ihrer Cluster erst mit 2, die fiktiven Cluster 0 und 1 sind reserviert. In der FAT stehen an ihrer Stelle, wie man oben sieht, einige mysteriöse Bytes, die in TOS keine Wirkung haben und deswegen auch Null sein können. Unter MSDOS ist das erste Byte das sogenannte *Media-Byte* und entspricht seinem Pendant im Bootsektor (siehe Floppykurs Teil II). Daß hier beim ST seltsamerweise nach dem Formatieren im Desktop eine $F7 steht, ist ein Grund, warum man ST-Disketten auf MSDOS-Maschinen doch nicht so ohne weiteres lesen kann.

Prinzipiell ist die FAT einfach zu verstehen. Im Diskettenverzeichnis steht für jede Datei ein Zeiger auf den ersten Cluster, den diese Datei belegt (siehe dazu auch Floppykurs, Teil II, über das Inhaltsverzeichnis der Diskette). Diesen Zeiger verwendet TOS, um an die betreffende Stelle der Blockbelegungstabelle zu springen. Dort hat nämlich jeder Cluster auf der Diskette einen eigens für ihn reservierten Platz in der FAT: Cluster 2 bekommt Platz Nr. 2, Cluster 500 hat Platz 500.

Nehmen wir an, in der Directory steht als erster Cluster die '2'. Im FAT-Eintrag für Cluster 2 steht eine weitere Clusternummer '3'. Aha, denken wir uns, die Datei setzt sich also auch in Cluster 3 fort. Mit dieser Drei springen wir in den FAT-Eintrag für Cluster 3. Dort finden wir wieder einen Verweis auf eine andere Clusternummer (nicht unbedingt auf Cluster 4).

Irgendwann ist auch die längste Datei am Ende: Dies wird in der Regel durch die fiktiven Clusternummern $FF8-$FFF angezeigt. Ein Wert von $FF0 bis $FF7 in einem FAT-Eintrag bedeutet, daß der betroffene Cluster beschädigt ist, also nicht belegt werden kann. Außerdem bedeutet ein Wert 0 in einem FAT-Eintrag, daß der Cluster frei ist.

Während das noch recht einsichtig klingt, werden Sie jetzt in der Praxis Ihre Schwierigkeiten haben. Denn auch die FAT-Einträge werden im Intel-Format abgespeichert, das heißt, niedrigstwerti- ges Byte zuerst. Dadurch ergibt sich in einer FAT ein ziemlich undurchsichtiger Wust, nämlich in unserem Beispiel:

F7 FF FF 03 40 00 05 F0 FF ...

(Bytenr: 0 1 2 3 4 5 6 7 8 ...)

MS-DOS und auch TOS füllen nämlich 16-Bit-Zahlen, die auf der Diskette im INTEL-Format stehen, mit den 12-Bit-Einträgen auf. Um das zu verstehen, lassen Sie uns das Konzept der *Über-Cluster* einführen. Das sind ganz einfach immer drei Bytes in der FAT. Der erste Über-Cluster in unserer Beispiel-FAT wäre dann F7 FF FF und interessiert uns nicht, weil er ja nur für die fiktiven Cluster 0 und 1 benutzt wird. Der zweite Über-Cluster ist schon interessanter:

```
03 40 00
++  +
```

Nun nehmen wir an, ein Directoryeintrag zeige auf den Clustereintrag Nummero 2, also auf den Anfang unseres zweiten Über-Clusters. Die mit "+" gekennzeichneten Nibbles (halbe Hexzahlen sozusagen) gehören zum ersten Cluster des Über-Clusters; da wir Intel-Format vor uns haben, muß die Null aus dem zweiten Byte noch vor die "03", also: 003, und schon haben wir die Nummer des nächsten Folgeclusters. Diese 003 zeigt auf den Clustereintrag 3 in der FAT, und der ist im nicht markierten Bereich des obigen Über-Clusters zu finden. Mit diesen nicht gekennzeichneten Nibbles verfährt man ähnlich wie vorhin, auch hier rutscht der höherwertige Teil ("00") nach vorne, so daß wir eine hübsche 004 erhalten.

Für diejenigen, die das Selbermachen lieben, ist noch wichtig herauszufinden, wo ein bestimmter Eintrag in der FAT steht, und wie man algorithmisch vorgeht. Dazu gibt es ein Kochrezept:

Man prüfe, ob die gesuchte Clusternummer gerade ist. Ist sie das, multipliziere man sie mit 1.5 und lese dann das *Wort* ab der errechneten Stelle. Also zum Beispiel für die Clusternummer 4:

4 ist gerade, mal 1.5 ergibt 6, also liest man ab Byte 6 der Beispiel-FAT das Wort $05F0. Das dreht man wegen des Intel-Formats um zu $F005 und ignoriert das oberste Nibble – ergibt $005.

Ungerade Clusternummern multipliziert man ebenfalls mit 1.5 und ignoriert den Nachkommaanteil. Beispiel: Clusternummer 3 mal 1.5 ergibt 4.5. Ohne Nachkommastellen also 4. Ab Byte 4 lesen wir das Wort $4000. Was kommt jetzt? Richtig, Byteschaufelei wegen Intel-Format und so (Ergebnis $0040). Von diesem Ergebnis fällt das unterste Nibble weg (entspricht einer Division durch 16 oder 4 Rechtsshifts): Macht $004.

Wenn Sie diesen Algorithmus nachvollziehen und mit dem

Konzept der Über-Cluster vergleichen, werden Sie feststellen, daß ein Über-Cluster immer an einem geraden Clustereintrag beginnt. Ist die Nummer des gesuchten Eintrags also gerade, greift man auf die mit "+" markierten Nibbles zu, ansonsten auf die anderen. Genau dieses Verhalten bildet der Algorithmus nach.

Ich denke, damit sind Sie fürs erste genügend verwirrt. Das war jetzt aber wirklich der vorläufig letzte Teil des Floppykurses; ich möchte mich jetzt meiner Festplatte widmen. Mal sehen, ob sich da ähnlich viel herausholen läßt wie aus der Floppy und deren Controller. Keep hacking, Ihr

Claus Brod



Der erweiterte Minimon

Listing 1:

```
  1:  '
  2:  ' Minimon für den ST
  3:  ' Written 1987 by Claus Brod
  4:  '                 Am Felsenkeller 2
  5:  '                 8772 Marktheidenfeld
  6:  '
  7:  ' V.2.1 mit rudimentärer Registerroutine
  8:  ' V.2.2 Registerroutine korrigiert,
      zusätzliche Fileausgabe
  9:  ' V.2.3 mit Speedtest
 10:  '
 11:  Cls
 12:  Do
 13:    Print
 14:    Print "Minimon ST - (C) 1987 by Claus Brod"
 15:    Print
 16:    Repeat
 17:      Print "Sektormonitor oder Trackmonitor
            oder Quit (S/T/Q)?"
 18:      A$=Upper$(Input$(1))
 19:    Until A$="S" Or A$="T" Or A$="Q"
 20:    Exit If A$="Q"
 21:    If A$="S"
 22:      @Sektormon
 23:    Else
 24:      @Trackmon
 25:    Endif
 26:  Loop
 27:  End
 28:  '
 29:  '
 30:  Procedure Gibmirzeit
 31:    Alert 1,"Funktion noch nicht|implemen-
        tiert.",1,"OK",A
 32:  Return
 33:  '
 34:  ' Prozedur Trackmon
 35:  ' initialisiert den Trackmonitor
 36:  ' zeigt in einer Schleife das Menü an, fragt
      auf Tastendruck ab
 37:  ' und verteilt auf die Unterroutinen
 38:  '
 39:  Procedure Trackmon
 40:    @Init_trackmon
 41:    Cls
 42:    Prn%=0
 43:    Richtung%=0
 44:    Seite%=0
 45:    Drive%=0
 46:    Steprate%=1            ! 3ms Steprate
 47:    R$=Chr$(27)+"p"
 48:    O$=Chr$(27)+"q"
 49:    @Rst                   ! auf Track 0 zurück-
                                 fahren
 50:    Do
 51:      Print
 52:      Print "******************* TRACKMON (C)
            1987 Claus Brod *******************"
 53:      @Prntable
 54:      Print R$;"0";O$;" Track 0, ";R$;"1";O$;
            "Step, ";R$;"2";O$;" Step-In, ";
 55:      Print R$;"3";O$;" Step-Out, ";R$;"4";O$;
            "Seek, ";R$;"5";O$;" Rd Sec, ";
 56:      Print R$;"6";O$;" Wr Sec"
 57:      Print R$;"7";O$;" Rd Adr, ";R$;"8";O$;
            "Rd Trk, ";R$;"9";O$;" Wr Trk, ";
 58:      Print R$;"A";O$;" IRQ, ";R$;"D";O$;"ump
            Buf, ";R$;"S";O$;"eite (";Seite%;
 59:      Print "), ";R$;"P";O$;"rn (o";
 60:      If Prn%=1
 61:        Print "n)"
 62:      Else
 63:        If Prn%=0
 64:          Print "ff)"
 65:        Else
 66:          Print "n file)"
 67:        Endif
 68:      Endif
 69:      Print R$;"R";O$;"egister, ";R$;"G";O$;
            "eschwindigkeit, ";R$;"Q";O$;"uit"
 70:      M$=Upper$(Input$(1))
 71:      Exit If M$="Q"
 72:      If M$="G"
 73:        @Speed
 74:      Endif
 75:      If M$="S"
 76:        Seite%=1-Seite%
 77:      Endif
 78:      If M$="P"
 79:        Inc Prn%
 80:        If Prn%>2
 81:          Prn%=0
 82:        Endif
 83:        If Prn%=2
 84:          Input "Filename";Fi$
 85:        Endif
 86:      Endif
 87:      If M$="D"
 88:        @Dump_buf
 89:      Endif
 90:      If M$="A"
 91:        @Irq                 ! FDC unterbrechen
 92:      Endif
 93:      If M$="R"
 94:        @Register
 95:      Endif
 96:      If M$<="9" And M$>="0"
 97:        On Val(M$)+1 Gosub Rst,Step,Step_in,
              Step_out,Seek,Rdsec,Wrsec,Rdadr
 98:        On Val(M$)-7 Gosub Rdtrk,Wrtrk
 99:      Endif
100:    Loop
101:  Return
102:  '
103:  '
104:  Procedure Init_trackmon
105:    Print "Einen Moment, bitte!"
106:    Restore Locksley
107:    @Readprog(1000)
108:    ' LOCKSLEY.S einlesen
109:    Inter$=Prg$
110:    Inter=Varptr(Inter$)
111:    Opcode=Inter+3
112:    Restore Select
113:    @Readprog(1000) ! Reserve für Erweiterung
114:    ' SELECT.S einlesen
115:    Sel$=Prg$
116:    Sel=Varptr(Sel$)
117:    Laufwerk=Sel+3
118:  Return
119:  '
120:  '
121:  Procedure Select
122:    Poke Laufwerk,Seite%+2 ! Laufwerk A,
                                 Seite 'Seite%'
123:    Call Sel
124:  Return
125:  Procedure Deselect
126:    Poke Laufwerk,0          ! deselektieren
127:    Call Sel
128:  Return
129:  '
130:  Procedure Mach_schon(O%)
131:    @Select            ! Laufwerk selektieren
132:    Poke Opcode,O%
133:    Call Inter         ! Kommando ausführen
134:    @Deselect          ! Laufwerk abwählen
135:  Return
136:  '
137:  Procedure Register
138:    ' sel+3:8 für lesen, 9 für schreiben,
        10 für DMA-Status
139:    ' in sel+4 wird das Register übergeben
140:    ' in sel+8 wird der Wert übergeben
141:    Print R$;"Register";O$
142:    Print R$;"L";O$;"esen oder ";R$;"S";O$;
          "chreiben?"
143:    Repeat
144:      A$=Upper$(Input$(1))
145:    Until A$="L" Or A$="S"
146:    If A$="L"
```

Listing 1: Fortsetzung

```
147:        @Get_them_all
148:      Else
149:        Op%=9
150:        Poke Sel+3,Op%
151:        Print "(1) Kommandoregister"
152:        Print "(2) Trackregister"
153:        Print "(3) FDC-Sektorregister"
154:        Print "(4) Datenregister"
155:        Print "(5) DMA-Sektorregister"
156:        Repeat
157:          A=Val(Input$(1))
158:        Until A>0 And A<6
159:        R%=(A-1)*2
160:        If A=5
161:          R%=16
162:        Endif
163:        Dpoke Sel+4,128+R%
164:        If Op%=9
165:          Input "Neuer Wert für das Register";W%
166:          Dpoke Sel+8,W%
167:        Endif
168:        Call Sel
169:        Print "Opcode:   $";Hex$(Peek(Sel+3))
170:        Print "Register: $";Hex$(Dpeek(Sel+4))
171:        Print "Wert:     $";Hex$(Dpeek(Sel+8) And
            255)
172:        Print
173:      Endif
174:    Return
175:    '
176:    '
177:    Procedure Get_them_all
178:      Poke Sel+3,8           ! Register lesen
179:      Dpoke Sel+4,128+0
180:      Call Sel
181:      Print "FDC-Statusregister: $";
          Hex$(Dpeek(Sel+8) And 255)
182:      Dpoke Sel+4,128+2
183:      Call Sel
184:      Print "Trackregister:      $";
          Hex$(Dpeek(Sel+8) And 255)
185:      Dpoke Sel+4,128+4
186:      Call Sel
187:      Print "FDC-Sektorregister: $";
          Hex$(Dpeek(Sel+8) And 255)
188:      Dpoke Sel+4,128+6
189:      Call Sel
190:      Print "Datenregister:      $";
          Hex$(Dpeek(Sel+8) And 255)
191:      Dpoke Sel+4,128+16
192:      Call Sel
193:      Print "DMA-Sektorregister: $";
          Hex$(Dpeek(Sel+8) And 255)
194:      Poke Sel+3,10   ! DMA-Status lesen
195:      Call Sel
196:      Print "DMA-Status:         $";
          Hex$(Dpeek(Sel+8) And 255)
197:    Return
198:    '
199:    ' Prozedur Rst
200:    ' fährt den Lesekopf auf Spur 0 zurück
201:    '
202:    Procedure Rst
203:      Print R$;"Restore";O$
204:      Track%=0
205:      Richtung%=0
206:      @Mach_schon(0+Steprate%) ! Restore-Befehl
207:    Return
208:    '
209:    ' Prozedur IRQ
210:    ' unterbricht den Floppycontroller bei der
      Arbeit
211:    '
212:    '
213:    Procedure Speed
214:      Local Dr,Lp%,Avr
215:      @Irq
216:      Cls
217:      Print At(5,2);"Drehzahlmessung"
218:      Print At(5,9);"    Drehzahl : "
219:      Print At(5,11);"Durchschnitt : "
220:      Mfp%=&HFFFA01
221:      Genau%=10
222:      @Select
223:      Dpoke Sel+2,9
224:      Dpoke Sel+4,&H80
225:      Dpoke Sel+8,&HD4
226:      Call Sel
227:      Repeat               ! Warte auf Index
228:      Until (Peek(Mfp%) And 32)=0 ! Dann Start !
229:      Repeat
230:        Dpoke Sel+2,9
231:        Dpoke Sel+4,&H80
232:        Dpoke Sel+8,&HD4
233:        T2=Timer
234:        For T%=1 To Genau%
235:          Call Sel
236:          Repeat
237:          Until (Peek(Mfp%) And 32)=0
238:          If Inkey$<>"" Or Mousek
239:            T%=99
240:          Endif
241:        Next T%
242:        Exit If T%=100
243:        Dr=12000*Genau%/(Timer-T2)
244:        Print At(22,9);Using "###.###",Dr
245:        Add Avr,Dr
246:        Inc Lp%
247:        Print At(22,11);Using "###.###",Avr/Lp%
248:      Until Inkey$<>"" Or Mousek
249:      @Irq
250:      @Deselect
251:    Return
252:    '
253:    '
254:    Procedure Irq
255:      Print R$;"IRQ";O$
256:      @Mach_schon(208)     ! IRQ-Befehl
257:    Return
258:    '
259:    ' Prozedur Step
260:    ' fährt den Lesekopf einen Schritt in die ein
      geschlagene Richtung
261:    '
262:    Procedure Step
263:      Print R$;"Step";O$
264:      Track%=Track%+Richtung%
265:      @Mach_schon(32+16+Steprate%) ! Step mit
                                       Track-Update
266:    Return
267:    '
268:    ' Prozedur Step_in
269:    ' fährt den Lesekopf einen Schritt nach innen
270:    '
271:    Procedure Step_in
272:      Print R$;"Step-in";O$
273:      Richtung%=1
274:      Track%=Track%+Richtung%
275:      @Mach_schon(64+16+Steprate%) ! Step-in mit
                                         Update
276:    Return
277:    '
278:    ' Prozedur Step-out
279:    ' fährt den Kopf einen Schritt nach außen
280:    '
281:    Procedure Step_out
282:      Richtung%=-1
283:      Print R$;"Step-out";O$
284:      Track%=Track%+Richtung%
285:      @Mach_schon(96+16+Steprate%) ! Step-out mit
                                         Update
286:    Return
287:    '
288:    ' Prozedur Seek
289:    ' fährt den Lesekopf auf die gewünschte Spur
290:    '
291:    Procedure Seek
292:      Print R$;"Seek";O$
293:      Input "Zieltrack";Trk
294:      If Trk>Track%
295:        Richtung%=1
296:      Else
297:        If Trk<Track%
298:          Richtung%=-1
299:        Endif
300:      Endif
301:      Track%=Trk
302:      Poke Inter+5,Trk  ! Spurnummer abliefern
303:      @Mach_schon(16+Steprate%) ! Seek-Befehl
304:    Return
305:    '
306:    ' Prozedur Rdsec
307:    ' liest Sektor(en) auf aktuellem Track ein
308:    '
309:    Procedure Rdsec
310:      Print R$;"Read sector";O$
311:      Input "Welcher Sektor";Sector%
312:      Poke Inter+7,Sector% ! Sektornummer
                               schreiben
313:      Input "Wieviel Bytes";Laenge
314:      Buf$=Space$(12*512)
315:      Dpoke Inter+8,Laenge ! Länge der Über-
                               tragung
316:      Lpoke Inter+10,Varptr(Buf$) ! Puffer-
                                      adresse
317:      @Mach_schon(128+16) ! Read multiple
                             sectors
318:    Return
319:    '
320:    ' Prozedur Rdadr
321:    ' Liest Adreßfelder auf aktuellem Track ein
322:    '
323:    Procedure Rdadr
324:      Print R$;"Read Adress";O$
325:      Input "Wieviele Adreßfelder";Laenge
326:      Dpoke Inter+8,Laenge ! Zahl der Adreßfelder
```

```
327:      Laenge=Laenge*6 ! 6 Byte pro Adreßfeld
328:      Buf$=Space$(512)
329:      Lpoke Inter+10,Varptr(Buf$) ! Pufferadresse
330:      Buf2$=Space$(100)
331:      Lpoke Inter+14,Varptr(Buf2$) ! Puffer-
                                        adresse für
                                        Status
332:     @Mach_schon(192) ! Rd-Address-Befehl
333:   Return
334:   '
335:   ' Prozedur Wrsec
336:   ' schreibt Sektor(en) auf aktuellen Track
337:   '
338:   Procedure Wrsec
339:     Print R$;"Write sector";O$
340:     Input "Welcher Sektor";Sek
341:     Poke Inter+7,Sek ! Sektornummer
342:     Input "Wieviel Bytes";Laenge
343:     Dpoke Inter+8,Laenge ! Länge der Über-
                                tragung
344:     Lpoke Inter+10,Varptr(Buf$) ! Puffer-
                                      adresse
345:     @Mach_schon(160+16) ! Write multiple
346:   Return                 sectors
347:   '
348:   ' Prozedur Rdtrk
349:   ' Aktuellen Track einlesen
350:   '
351:   Procedure Rdtrk
352:     Print R$;"Read Track";O$
353:     Buf$=Space$(8000)
354:     Lpoke Inter+10,Varptr(Buf$) ! Pufferadresse
355:     Input "Wieviel Bytes";Laenge
356:     Dpoke Inter+8,Laenge ! Länge der
                                Übertragung
357:     @Mach_schon(192+32) ! Read Track
358:   Return
359:   '
360:   ' Prozedur Wrtrk
361:   ' Einen Track formatieren
362:   '
363:   Procedure Wrtrk
364:     Print R$;"Write Track";O$
365:     T$=Space$(8000)
366:     ' liest die Trackdaten ab dem Label
          Trackdaten ein
367:     ' Format der Datazeilen:
368:     ' zu schreibendes Byte, danach Anzahl (wie
369:     ' oft soll dieses Byte geschrieben werden)
370:     ' "***,***" heißt Schluß
371:     Restore Trackdaten
372:     Cn=1
373:     Do
374:       Read B$,Z$
375:       @Hextodec(B$)
376:       B=S
377:       @Hextodec(Z$)
378:       Z=S
379:       Exit If B$="***"
380:       If Upper$(B$)="TRACK"
381:         B=Track%
382:       Endif
383:       If Upper$(B$)="SIDE"
384:         B=Seite%
385:       Endif
386:       For T=1 To Z
387:         Print B'
388:         Mid$(T$,Cn,1)=Chr$(B)
389:         Inc Cn
390:       Next T
391:     Loop
392:     Print
393:     Lpoke Inter+10,Varptr(T$) ! Pufferadresse
394:     Input "Wieviel Bytes";Laenge
395:     Dpoke Inter+8,Laenge ! Länge der Über-
                                tragung
396:     @Mach_schon(15*16)   ! Write Track
397:   Return
398:   '
399:   ' Prozedur Dump_buf
400:   ' Gibt ersten oder zweiten Puffer aus
401:   ' und erlaubt ihn zu edieren
402:   '
403:   Procedure Dump_buf
404:     Repeat
405:       Print "Ersten oder zweiten Puffer
            ausgeben (1/2)?"
406:       Ch$=Input$(1)
407:     Until Ch$="1" Or Ch$="2"
408:     If Ch$="1"
409:       Sec$=Buf$
410:       @Dump(Laenge)
411:     Else
412:       Sec$=Buf2$
413:       @Dump(Len(Buf2$))
414:     Endif
415:     Print R$;"Edit (Y/N)?";O$
416:     A$=Upper$(Input$(1))
417:     If A$="Y"
418:       If Ch$="1"
419:         Sec$=Buf$
420:         @Edit
421:         Buf$=Sec$
422:       Else
423:         Sec$=Buf2$
424:         @Edit
425:         Buf2$=Sec$
426:       Endif
427:     Endif
428:   Return
429:   '
430:   ' Prozedur Prntable
431:   ' Statusmeldung ausgeben
432:   '
433:   Procedure Prntable
434:     Inter=Varptr(Inter$)
435:     Opcode=Inter+3
436:     Sel=Varptr(Sel$)
437:     Laufwerk=Sel+3
438:     Print "Opcode                    :";
          Dpeek(Inter+2);Tab(40);
439:     If Dpeek(Inter+30)=0
440:       Print "Kein ";
441:     Endif
442:     Print "Timeout!"
443:     Print "Track                     :";
          Track%;Tab(40);
444:     Print "Sektor                    :";
          Dpeek(Inter+6)
445:     Print "Adresse des Puffers 1     :";
          Lpeek(Inter+10);Tab(40);
446:     Print "Adresse des Puffers 2     :";
          Lpeek(Inter+14)
447:     Print "FDC-Status                :";
          Peek(Inter+19);Tab(40);
448:     Print "DMA-Status                :";
          Peek(Inter+21)
449:     Print "Startadresse DMA          :";
          Lpeek(Inter+22);Tab(40);
450:     Print "Endadresse DMA            :";
        : Lpeek(Inter+26)
451:     Print "Gelesene/geschriebene Bytes:";
          Dpeek(Inter+8)
452:   Return
453:   '
454:   ' Prozedur Readprog
455:   ' Liest Programm aus Datazeilen ein
456:   '
457:   Procedure Readprog(L)
458:     Prg$=Space$(L)
459:     T=1
460:     Repeat
461:       Read A$
462:       If A$<>"***"
463:         Mid$(Prg$,T,1)=Chr$(Val("&h"+A$))
464:       Endif
465:       Inc T
466:     Until A$="***"
467:   Return
468:   '
469:   ' Prozedur Hextodec
470:   ' Wandelt A$ in Dezimal um (Ziel: S)
471:   '
472:   Procedure Hextodec(A$)
473:     S=0
474:     A$=Upper$(A$)
475:     For I=1 To Len(A$)
476:       A=Asc(Right$(A$,I))
477:       If A>64
478:         A=A-7
479:       Endif
480:       A=A-48
481:       S=S+A*16^(I-1)
482:     Next I
483:   Return
484:   '
485:   '
486:   ' Sektormon: Kleiner Diskmon mit Standard-
        fähigkeiten
487:   '
488:   Procedure Sektormon
489:     Cls
490:     Prn%=0
491:     Status%=0
492:     Seite%=0
493:     Track%=0
494:     Sector%=1
495:     Drive%=0
496:     Sec$=Space$(512)
497:     R$=Chr$(27)+"p"
498:     O$=Chr$(27)+"q"
499:     @Lesen
500:     Do
501:       Print
502:       Print "****** Minimon (C) 1987 Claus
          Brod *** Status: ";Status%;" *****"
503:       Print R$;"R";O$;"ead, ";R$;"W";O$;"rite,
          ";R$;"T";O$;"rk (";Track%;"), ";
504:       Print R$;"S";O$;"eite (";Seite%;"), Se";
          R$;"k";O$;"tor (";Sector%;"), ";
505:       Print R$;"E";O$;"dit, ";R$;"D";O$;
          "ump, ";R$;"P";O$;"rn (o";
```

```
506:       If Prn%
507:         Print "n), ";
508:       Else
509:         Print "ff), ";
510:       Endif
511:       Print "E";R$;"x";O$;"ec, ";R$;"Q";O$;
           "uit"
512:      M$=Upper$(Input$(1))
513:      If M$="X"
514:        @Exe
515:      Endif
516:      If M$="P"
517:        Prn%=1-Prn%
518:      Endif
519:      If M$="T"
520:        Input "Track";Track%
521:      Endif
522:      If M$="S"
523:        Input "Seite";Seite%
524:      Endif
525:      If M$="K"
526:        Input "Sektor";Sector%
527:      Endif
528:      If M$="R"
529:        @Lesen
530:      Endif
531:      If M$="W"
532:        @Schreiben
533:      Endif
534:      If M$="E"
535:        @Edit
536:      Endif
537:      If M$="D"
538:        @Dump(512)
539:      Endif
540:      Exit If M$="Q"
541:    Loop
542:  Return
543:  '
544:  '
545:  ' Sektor einlesen
546:  '
547:  Procedure Lesen
548:    Buffer%=Varptr(Sec$)
549:    Status%=Xbios(8,L:Buffer%,L:0,Drive%,
        Sector%,Track%,Seite%,1)
550:  Return
551:  '
552:  '
553:  ' Sektor schreiben
554:  '
555:  Procedure Schreiben
556:    Buffer%=Varptr(Sec$)
557:    Status%=Xbios(9,L:Buffer%,L:0,Drive%,
        Sector%,Track%,Seite%,1)
558:  Return
559:  '
560:  '
561:  ' Sektorpuffer ausgeben
562:  '
563:  Procedure Dump(L)
564:    If Prn%=0
565:      Open "O",#1,"con:"
566:    Else
567:      If Prn%=1
568:        Open "O",#1,"prn:"
569:      Else
570:        Open "O",#1,Fi$
571:      Endif
572:    Endif
573:    Print #1,"Track: ";Track%;" Sektor: ";
        Sector%;" Seite: ";Seite%
574:    For T=1 To L Step 16
575:      A$=Str$(T-1)
576:      While Len(A$)<3
577:        A$="0"+A$
578:      Wend
579:      D$="  "
580:      Print #1,A$;D$;
581:      For I=0 To 15
582:        V$=Mid$(Sec$,T+I,1)
583:        A$=Hex$(Asc(V$))
584:        If V$<" " Or V$>"z"
585:          V$="."
586:        Endif
587:        Print #1,Right$("0"+A$,2)'
588:        D$=D$+V$
589:      Next I
590:      Print #1,D$
591:      A$=Inkey$
592:      If A$=" "
593:        A=Inp(2)
594:      Endif
595:      Exit If A$>"" And A$<>" "
596:    Next T
597:    Close #1
598:  Return
599:  '
600:  '
601:  ' Byte ändern
602:  '
603:  Procedure Edit
604:    Input "Byte Nr.";Byte%
605:    Print "Alter Wert ist: ";Asc(Mid$(Sec$,
        Byte%+1,1))
606:    Input "Neuer Wert in dezimal oder ASCII";
        Wert$
607:    Wert%=Val(Wert$)
608:    If Wert%=0 And Left$(Wert$,1)<>"0"
609:      Wert%=Asc(Wert$)
610:    Endif
611:    Mid$(Sec$,Byte%+1,1)=Chr$(Wert%)
612:  Return
613:  '
614:  '
615:  ' anderes Programm ausführen
616:  '
617:  Procedure Exe
618:    Fileselect "*.*","",Sel$
619:    If Exist(Sel$)<>0
620:      S=Fre(0)
621:      Reserve 50000
622:      Showm
623:      Exec 0,Sel$,"",""
624:      Reserve S-1000
625:    Endif
626:  Return
627:  '
628:  '
629:  ' Datawüste mit den Daten für LOCKSLEY und
630:  ' SELECT sowie ein paar Trackdaten
631:  '
632:  Locksley:
633:  Data 60,24,0,0,0,0,0,0,0,0,0,0,0,0,0,0
634:  Data 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
635:  Data 0,0,0,0,0,0,48,E7,FF,FE,42,80,61,0,1,36
636:  Data 45,FA,FF,EE,24,80,3C,3A,FF,CA,45,FA,FF,
       E8,34,BC
637:  Data 0,0,45,FA,FF,DA,34,BC,0,0,50,F9,0,0,4,3E
638:  Data 61,0,1,1E,51,F9,0,0,4,3E,45,FA,FF,C4,20,
       12
639:  Data 61,0,1,2,4C,DF,7F,FF,4E,75,32,3C,0,1E,
       61,A
640:  Data 33,C7,0,FF,86,4,32,3C,0,1E,51,C9,FF,FE,
       4E,75
641:  Data 2E,3C,0,4,0,0,8,39,0,5,0,FF,FA,1,67,60
642:  Data 53,87,67,40,45,FA,FF,8E,4A,52,67,EA,45,
       FA,FF,70
643:  Data 15,79,0,FF,86,9,0,1,15,79,0,FF,86,B,0,2
644:  Data 15,79,0,FF,86,D,0,3,20,3A,FF,54,B0,BA,
       FF,5C
645:  Data 6D,C4,1E,3C,0,D0,61,0,0,EC,45,FA,FF,58,
       34,BC
646:  Data 0,0,60,1C,61,1A,1E,3C,0,D0,61,0,0,D8,45,
       FA
647:  Data FF,3E,34,BC,0,1,45,FA,FF,3C,34,BC,0,0,
       4E,75
648:  Data 30,39,0,FF,86,4,45,FA,FF,1A,34,80,4E,75,
       30,39
649:  Data 0,FF,86,6,45,FA,FF,E,34,80,42,41,12,39,
       0,FF
650:  Data 86,9,E1,89,12,39,0,FF,86,B,E1,89,12,39,
       0,FF
651:  Data 86,D,45,FA,FE,F6,24,81,92,BA,FE,EC,45,
       FA,FE,DA
652:  Data 34,81,4E,75,45,FA,FE,E0,24,87,13,C7,0,
       FF,86,D
653:  Data E0,8F,13,C7,0,FF,86,B,E0,8F,13,C7,0,FF,
       86,9
654:  Data 2E,3A,FE,C4,42,80,30,3A,FE,B0,DE,80,45,
       FA,FE,BC
655:  Data 24,87,4E,75,2F,0,3F,3C,0,20,4E,41,5C,8F,
       4E,75
656:  Data 3E,6,3A,6,8,6,0,7,66,10,CC,3C,0,F0,BC,3C
657:  Data 0,10,67,0,1,64,66,0,1,50,8,6,0,6,66,A
658:  Data 8,6,0,5,67,2A,66,0,0,F4,CC,3C,0,F0,BC,3C
659:  Data 0,C0,67,0,1,66,BC,3C,0,E0,67,6A,BC,3C,0,
       F0
660:  Data 67,0,0,9E,61,0,FE,B4,32,3C,0,FA,60,0,FE,
       BC
661:  Data 2E,3A,FE,48,61,0,FF,6E,45,FA,FE,5A,34,
       BC,0,1
662:  Data 33,FC,0,90,0,FF,86,6,33,FC,1,90,0,FF,86,
       6
663:  Data 33,FC,0,90,0,FF,86,6,3E,3C,0,E,61,0,FE,
       7C
664:  Data 33,FC,0,84,0,FF,86,6,3E,3A,FE,C,61,0,FE,
       6C
665:  Data 33,FC,0,80,0,FF,86,6,3E,5,61,0,FE,5E,61,
       0
666:  Data FE,70,60,0,FE,EA,2E,3A,FD,F2,61,0,FF,18,
       45,FA
667:  Data FE,4,34,BC,0,1,33,FC,0,90,0,FF,86,6,33,
       FC
668:  Data 1,90,0,FF,86,6,33,FC,0,90,0,FF,86,6,3E,
       3C
669:  Data 0,E,61,0,FE,26,33,FC,0,80,0,FF,86,6,60,
       B8
670:  Data 2E,3A,FD,B8,61,0,FE,DE,45,FA,FD,CA,34,
       BC,0,1
671:  Data 33,FC,1,90,0,FF,86,6,33,FC,0,90,0,FF,
       86,6
672:  Data 33,FC,1,90,0,FF,86,6,3E,3C,0,E,61,0,FD,
       EC
```

```
673:   Data 33,FC,1,80,0,FF,86,6,60,0,FF,7E,2E,3A,
       FD,7C
674:   Data 61,0,FE,A2,45,FA,FD,8E,34,BC,0,1,33,FC,
       1,90
675:   Data 0,FF,86,6,33,FC,0,90,0,FF,86,6,33,FC,1,
       90
676:   Data 0,FF,86,6,3E,3C,0,E,61,0,FD,B0,33,FC,1,
       84
677:   Data 0,FF,86,6,3E,3A,FD,40,61,0,FD,A0,33,FC,
       1,80
678:   Data 0,FF,86,6,60,0,FF,32,33,FC,0,80,0,FF,
       86,6
679:   Data 61,0,FD,88,60,0,FD,9A,33,FC,0,86,0,FF,
       86,6
680:   Data 3E,3A,FD,12,61,0,FD,74,33,FC,0,80,0,FF,
       86,6
681:   Data 3E,5,61,0,FD,66,60,0,FD,78,2E,3A,FC,FE,
       26,7A
682:   Data FC,FE,61,0,FE,20,33,FC,0,90,0,FF,86,6,
       33,FC
683:   Data 1,90,0,FF,86,6,33,FC,0,90,0,FF,86,6,3E,
       3C
684:   Data 0,1,61,0,FD,36,33,FC,0,80,0,FF,86,6,34,
       3A
685:   Data FC,C8,3E,5,61,0,FD,24,61,0,FD,36,32,3A,
       FC,C4
686:   Data 36,C1,51,CA,FF,EE,60,0,FD,A6,0,0
687:   Data ***
688:   '
689:   Select:
690:   Data 60,E,0,0,0,0,0,0,0,0,0,0,0,0,0,0
691:   Data 48,E7,FF,FE,42,80,61,6A,45,FA,FF,F2,24,
       80,50,F9
692:   Data 0,0,4,3E,3E,3A,FF,DC,66,14,33,FC,0,80,0,
       FF
693:   Data 86,6,32,39,0,FF,86,4,8,1,0,7,66,F4,C,7
694:   Data 0,8,6C,5C,A,7,0,7,CE,3C,0,7,40,E7,0,7C
695:   Data 7,0,13,FC,0,E,0,FF,88,0,10,39,0,FF,88,0
696:   Data C0,3C,0,F8,8E,0,13,C7,0,FF,88,2,51,F9,0,
       0
697:   Data 4,3E,46,DF,45,FA,FF,96,20,12,61,6,4C,DF,
       7F,FF
698:   Data 4E,75,2F,0,3F,3C,0,20,4E,41,5C,8F,4E,75,
       51,C9
699:   Data FF,FE,4E,75,2F,A,3F,3C,0,9,4E,41,5C,8F,
       4E,75
700:   Data C,7,0,8,67,14,C,7,0,9,67,20,C,7,0,A
701:   Data 67,52,45,FA,0,68,61,DC,60,BA,33,FA,FF,
       48,0,FF
702:   Data 86,6,61,1C,45,FA,FF,42,34,80,60,A8,3E,
       3A,FF,3A
703:   Data CE,7C,0,FF,33,FA,FF,2E,0,FF,86,6,61,14,
       60,94
704:   Data 32,3C,0,1E,61,A8,30,39,0,FF,86,4,32,3C,
       0,1E
705:   Data 60,9C,32,3C,0,1E,61,96,33,C7,0,FF,86,4,
       32,3C
706:   Data 0,1E,60,8A,33,FC,0,90,0,FF,86,6,30,39,0,
       FF
707:   Data 86,6,45,FA,FE,F4,34,80,60,0,FF,5A,52,6F,
       75,74
708:   Data 69,6E,65,20,6E,6F,63,68,20,6E,69,63,68,
       74,20,69
709:   Data 6D,70,6C,65,6D,65,6E,74,69,65,72,74,2E,
       D,A,0
710:   Data 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
711:   Data ***
712:   '
713:   Trackdaten:
714:   Data 4E,2
715:   ' Trackvorspann
716:   '
717:   Data 00,2
718:   ' Sektorvorspann
719:   Data F5,3,FE,1
720:   ' Syncbytes und Adreßmarke
721:   Data Track,1,Side,1,1,1,3,1,F7,1
722:   ' Sektorvorspann und Checksumme
723:   Data 4E,14,0,A
724:   ' Lückenbytes
725:   Data F5,3,FB,1
726:   ' Syncbytes und Datenmarke
727:   Data CB,400,F7,1
728:   ' Sektordaten und Checksumme
729:   Data 4E,10
730:   ' Lückenbytes
731:   '
732:   Data 00,2
733:   ' Sektorvorspann
734:   Data F5,3,FE,1
735:   ' Syncbytes und Adreßmarke
736:   Data Track,1,Side,1,2,1,0,1,F7,1
737:   ' Sektorheader (Sektor mit 128 Bytes) und
       Checksumme
738:   Data 4E,14,0,A
739:   ' Lückenbytes
740:   Data F5,3,FB,1
741:   ' Syncs und Datenmarke
742:   Data CB,40,F7,1
743:   ' Sektordaten (unvollständig) und Checksumme
744:   Data 4E,10
745:   ' Lückenbytes
746:   '
747:   Data 00,2
748:   ' Sektorvorspann
749:   Data F5,3,FE,1
750:   Data Track,1,Side,1,3,1,2,1,F7,1
751:   ' normaler 512-Byte-Sektor
752:   Data 4E,14,0,A
753:   Data F5,3,FB,1
754:   Data CB,200
755:   ' Sektordaten
756:   Data F7,1,4E,10
757:   '
758:   Data 00,2
759:   ' Sektorvorspann
760:   Data F5,3,FE,1
761:   Data Track,1,Side,1,4,1,4,1,F7,1
762:   ' Sektor mit der Größenangabe '4'
763:   Data 4E,14,0,A
764:   Data F5,3,FB,1
765:   Data 02,200
766:   ' Sektordaten des zweiten Sektors Nr. 3
767:   Data F7,1,4E,10
768:   '
769:   Data 4E,200
770:   ' Tracknachspann
771:   Data ***,***
```

AUF TOUREN GEBRACHT

NOCH EINE ERWEITERUNG ZUM MINIMON

In den letzten beiden Folgen haben Sie ja schon miterlebt, daß der MINIMON sich ganz gut dazu eignet, selbstgebastelte Erweiterungen anzufügen. Um die Routine in Listing 1 einzubinden, müssen Sie nur das Menü des Trackmon-Teils etwas erweitern (einen Vorschlag dazu finden Sie ebenfalls im Listing). Die Routine mißt die Drehzahl des selektierten Laufwerks und gibt sie auf dem Schirm aus. Ein Tastendruck bricht die Messung ab.

Wie funktioniert die Routine? Das Programm schickt jeweils FORCE-IRQ-Befehle (siehe Floppykurs, Teil 5) an den Controller, deren Bit 2 gesetzt ist. Dadurch wird bei jedem Indeximpuls ein Interrupt ausgelöst. Dieser Interrupt wird dem MFP68901 im ST durchgemeldet. Der legt daraufhin Bit 5 seines GPIP-Registers ($FFFA01) auf Low-Pegel (löscht es also). Auf solche Low-Pegel wartet das Programm; nach 10 Indeximpulsen wird die Zeit gemessen, die das Laufwerk dafür gebraucht hat. Aus dieser Zeit kann man durch eine einfache Formel errechnen, wieviele Umdrehungen das Laufwerk hat.

Die Formel sollte ich vielleicht erklären. Sie lautet:

(1) Drehzahl = 12000 * Genau% / (Zeit für Genau% Um-

drehungen in 200stel Sekunden)

Versuchen wir uns dieser Formel schrittweise zu nähern. Bei Solldrehzahl – 300 Umdrehungen pro Minute – braucht das Laufwerk 0.2 s für eine Umdrehung. Das entspricht 40/200 Sekunden (diese Umrechnung brauchen wir gleich noch). Die tatsächliche Drehzahl ergibt sich nun aus dem Quotienten aus dieser Solldrehzeit und der wirklich gemessenen Zeit, multipliziert mit 300:

(2) Drehzahl = 300*Solldrehzeit/(gemessene Zeit)

Die Solldrehzeit ist (0.2 Sekunden * Genau%) = (40/200 Sekunden * Genau%). Setzt man das ein, ergibt sich:

(3) Drehzahl = 300*(40/200)*Genau%/(gemessene Zeit) = 12000*Genau% / ((gemessene Zeit)*200)

Damit sind wir schon ziemlich genau bei der ersten Formel angekommen. Den Faktor 200 im Nenner können wir uns schenken, weil im GFABASIC-Programm die Zeit eh schon in 200stel Sekunden (über die TIMER-Variable) gemessen wird. Damit steht genau die Formel (1) vor uns.

Die Messung ist hinreichend genau, in der abgedruckten Version liegt der Meßfehler bei maximal +/–0.7 Umdrehungen. Mit dem Parameter Genau% können Sie – falls Sie so pingelig sind – die Exaktheit verbessern. Für HYPERFORMAT-Disketten sind Umdrehungsgeschwindigkeiten von 300 bis 302 Upm optimal.

Nach dem Anlaufen der Routine sollten Sie eine Weile warten, bis sich die Drehzahl stabilisiert hat. Probieren Sie doch mal, was passiert, wenn Sie Ihr Laufwerk während des Drehzahltests drehen und schütteln (aber nicht übertreiben).

Trackmon mit eingebauter Speed-Routine

Listing

```
  1:   ' Diese  Anpassung der Routine Trackmon soll  Ihnen  ver-
  2:   ' deutlichen, wie  man  die  SPEED-Routine in  den  EX-
  3:   ' TENDED  MINIMON  ein bauen könnte.
  4:
  5:
  6:   Procedure Trackmon
  7:     @Init_trackmon
  8:     Cls
  9:     Prn%=0
 10:     Richtung%=0
 11:     Seite%=0
 12:     Drive%=0
 13:     Steprate%=1
 14:     R$=Chr$(27)+"p"
 15:     O$=Chr$(27)+"q"
 16:     @Rst                    ! auf Track 0 zurückfahren
 17:     Do
 18:       Print
 19:       Print "******************* TRACKMON (C) 1987 Claus Brod
             ********************"
 20:       @Prntable
 21:       Print R$;"0";O$;" Track 0, ";R$;"1";O$;" Step, ";R$;"2";
             O$;" Step-In, ";
 22:       Print R$;"3";O$;" Step-Out, ";R$;"4";O$;" Seek, ";R$;"5"
             ;O$;" Rd Sec, ";
 23:       Print R$;"6";O$;" Wr Sec"
 24:       Print R$;"7";O$;" Rd Adr, ";R$;"8";O$;" Rd Trk, ";R$;"9"
             ;O$;" Wr Trk, ";
 25:       Print R$;"A";O$;" IRQ, ";R$;"D";O$;"ump Buf, ";R$;"S";O$
             ;"eite (";Seite%;
 26:       Print "), ";R$;"P";O$;"rn (o";
 27:       If Prn%=1
 28:         Print "n)"
 29:       Else
 30:         If Prn%=0
 31:           Print "ff)"
 32:         Else
 33:           Print "n file)"
 34:         Endif
 35:       Endif
 36:       '
 37:       .
 38:       ' ............'Die folgende Zeile ist neu!!!!!!!!!!!!
 39:       Print R$;"R";O$;"egister, ";R$;"G";O$;"eschwindigkeit, "
             ;R$;
 40:       Print "Q";O$;"uit"
 41:       ' ............'weiter wie gewohnt
 42:       .
 43:       M$=Upper$(Input$(1))
 44:       Exit If M$="Q"
 45:       '
 46:       ' Jetzt wieder etwas Neues!!!!!!!!!!!!!!!!1
 47:       If M$="G"
 48:         @Speed
 49:       Endif
 50:       ' ab jetzt wieder kalter Kaffee
 51:       .
 52:       If M$="S"
 53:         Seite%=1-Seite%
 54:       Endif
 55:       If M$="P"
 56:         Inc Prn%
 57:         If Prn%>2
 58:           Prn%=0
 59:         Endif
 60:         If Prn%=2
 61:           Input "Filename";Fi$
 62:         Endif
 63:       Endif
 64:       If M$="D"
 65:         @Dump_buf
 66:       Endif
 67:       If M$="A"
 68:         @Irq                                ! FDC unterbrechen
 69:       Endif
 70:       If M$="R"
 71:         @Register
 72:       Endif
 73:       If M$<="9" And M$>="0"
 74:         On Val(M$)+1 Gosub Rst,Step,Step_in,Step_out,Seek,
               Rdsec,Wrsec,Rdadr
 75:         On Val(M$)-7 Gosub Rdtrk,Wrtrk
 76:       Endif
 77:     Loop
 78:   Return
 79:   '
 80:
 81:   Hier folgt jetzt die eigentliche Meßroutine SPEED.
 82:   !
 83:   :
 84:   ' Speed-Routine; mißt die Drehzahl eines Laufwerks
 85:   Procedure Speed
 86:     Local Dr,Lp%,Avr
 87:     @Irq
 88:     ' FDC in definierten Zustand
 89:     .
 90:     Cls
 91:     Print At(5,2);"Drehzahlmessung"
 92:     Print At(5,9);"    Drehzahl : "
 93:     Print At(5,11);"Durchschnitt : "
 94:     Mfp%=&HFFFA01
 95:     ' Hier meldet der MFP den FDC-Interrupt beim Indeximpuls
 96:     .
 97:     Genau%=10
 98:     ' Genauigkeitsfaktor, je höher, desto länger wird gemessen
 99:     .
100:     @Select
101:     ' Laufwerk selektieren
102:     Dpoke Sel+2,9
103:     Dpoke Sel+4,&H80
104:     Dpoke Sel+8,&HD4
105:     Call Sel
106:     ' INDEX-IRQ ($D4) an den FDC schicken
107:     ' (erweiterte LOCKSLEY-Routine wird benötigt, siehe
           ST 11/87)
108:     '
109:     Repeat                                ! Warte auf Index
110:     Until (Peek(Mfp%) And 32)=0           ! Dann Start !!
111:     Repeat
112:       Dpoke Sel+2,9
113:       Dpoke Sel+4,&H80
114:       Dpoke Sel+8,&HD4
115:       ' Parameterfeld der Select-Routine vorbereiten
116:       .
117:       T2=Timer
118:       For T%=1 To Genau%
119:         Call Sel
120:         Repeat
121:         Until (Peek(Mfp%) And 32)=0
122:         If Inkey$<>"" Or Mousek
123:           T%=99
124:         Endif
125:       Next T%
126:       ' Zeit für Genau% Umdrehungen messen
127:       ' Abbruch durch Taste oder Mausklick
128:       Exit If T%=100
129:       Dr=12000*Genau%/(Timer-T2)
130:       ' Formel für Drehzahl, siehe Artikel)
131:       Print At(22,9);Using "###.###",Dr
132:       Add Avr,Dr
133:       Inc Lp%
134:       Print At(22,11);Using "###.###",Avr/Lp%
135:     Until Inkey$<>"" Or Mousek
136:     @Irq
137:     @Deselect
138:   Return
```

- ★ kurze Einarbeitungszeit
- ★ dialogorientiert
- ★ menügesteuert
- ★ GEM Oberfläche
- ★ übersichtliche Eingabebildschirme
- ★ kein Hardware-Kopierschutz, kein Port belegt
- ★ **Auswertungen jederzeit in Sekundenschnelle auf Bildschirm, Drucker oder Massenspeicher**
- ★ frei wählbares Wirtschaftsjahr
- ★ variable Mwst.-Sätze
- ★ **integriertes Abschreibungsprogramm**
- ★ Journal, Saldenliste
- ★ Kassenbuch, Finanzkonten
- ★ Kosten- und Erlöselisten
- ★ Wareneingangsliste
- ★ **USTvoranmeldung direkt auf das amtliche Formular**
- ★ Einnahme-Überschuß-Rechnung (fibuMAN e/m)
- ★ **Bilanz und Gewinn- und Verlustrechnung nach dem Bilanzrichtliniengesetz (fibuMAN f/m)**
- ★ BWA — Betriebswirtschaftliche Auswertung (fibuMAN m, optional fibuMAN f)
- ★ Sammeldruck aller zum Monatsabschluß relevanter Auswertungen
- ★ Kontoauszüge über das gesamte Jahr nach Konten chronologisch geordnet
- ★ Hinweis auf unsinnige Buchungen
- ★ **auf Mausklick automatisches Anlegen einer Stornobuchung**
- ★ Automatikjournal zur einmaligen Eingabe monatlich wiederkehrender Buchungen
- ★ diverse Schnittstellen (fibuSTAT, Faktura in Vorbereitung: faktuMAN)
- ★ schnelles Suchen nach Buchungen mit optionaler Übernahme in das aktuelle Journal, z. B. Ausbuchungen von offenen Posten
- ★ Kontenanlage mit optionalem Automatiktext und -UST auch während der Buchungen
- ★ Hochrechnung der Mwst. von Nettobeträgen
- ★ Kennwort auf 2 Ebenen
- ★ Sicherheitskopien nach frei wählbarem Zeitraum
- ★ **komfortable Druckeranpassung**
- ★ Ordner vom Programm aus anzulegen
- ★ Dateien vom Programm aus zu löschen

Ich bin mit keinem Buchhaltungs-Programm zufrieden!!!

Ich brauche Zuverlässigkeit, Schnelligkeit, Übersichtlichkeit, aussagekräftige, gesetzlich anerkannte Auswertungen

und habe keine Zeit, mich erst lange einarbeiten zu müssen...



fibuMAN

BY H.-G. SIEGEL

DER FINANZBUCHHALTUNGS-MANAGER

NEU VERSION 2.0

Siehe Testbericht ST-Computer 12/87

fibuMAN m	fibuMAN f	fibuMAN e	BWA ZU fibuMAN f DM 98,-
MANDANTENFÄHIGE FIBU MIT BWA	FINANZ-BUCHHALTUNG	EINNAHME-ÜBERSCHUSS-RECHNUNG	DEMODISKETTE MIT HANDBUCH
DM 968,-	DM 768,-	398,-	DM 65,-

PRODATA bietet Ihnen vollen professionellen Anwenderservice wie kostenlose telefonische Hotline, täglich von 10 – 23 Uhr, Schulung und individuelle Einarbeitung, ggf. Anpassung der Programme an Ihre Bedürfnisse, prompten Updateservice und, und, und...
Ein Aufstieg in der fibuMAN-Reihe ist jederzeit unter Anrechnung des Preises der ursprünglichen Version möglich. Sie können alle hier aufgeführten Produkte direkt beim Entwickler PRODATA oder im autorisierten Fachhandel erwerben.

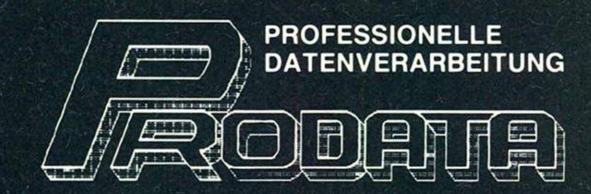
fibuKURS .... DM 348,-

- ★ Diskette mit ausführlichem Lehrbuch (weit über 100 Seiten)
- ★ leichter Einstieg in eine eigene EDV-Buchführung
- ★ Finanzbuchhaltung und Einnahme-Überschuß-Rechnung
- ★ Nachschlagewerk für das spätere praktische Buchen
- ★ Bibliothek von weit über 100 anschaulichen Fallbeispielen
- ★ nach Abschluß des Kurses ist selbständiges Arbeiten möglich

fibuSTAT .... DM 398,-

- ★ unabhängiges GEM-Statistikmodul zu allen fibuMAN Programmen
- ★ Berechnung von betriebswirtschaftlich interessanten Kennziffern
- ★ diverse grafisch vergleichende Statistiken
- ★ übersichtliche Grafiken für einen schnellen Einblick in Ihre jeweilige Finanzlage
- ★ Balken- und Kuchendiagramme

BROICHERSTRASSE 39
5060 BERGISCH GLADBACH 1
TEL. 0 22 04 / 5 14 56 – 01 61 / 221 57 91



ENTWICKLUNG UND VERTRIEB INTELLIGENTER ANWENDERSYSTEME

Programmier Praxis

Es ist nun das dritte Mal, daß diese feste Spalte bei uns erscheint und wir müssen feststellen, daß die Reaktion unserer Leser noch größer ist, wie wir gedacht hatten. Langsam könnten wir wesentlich mehr Seiten mit solchen Programmen füllen. Doch auch unsere nicht programmierenden Leser haben ein Recht auf Platz in der ST-Computer.

Vergessen Sie nicht, daß diese Rubrik mit Ihrer Hilfe gestaltet wird. Also senden Sie uns Ihre Vorschlage. Bis zum nächsten Mal.

INHALT



Der verlängerte Arm

– ein Software-Reset für alle ST's

Das Programm ist hauptsächlich für diejenigen gedacht, die sich unter dem Motto "Auf zu neuen Ufern" leichtsinnigerweise einen Mega ST gekauft haben, ohne rechtzeitig zu bemerken, daß die schönste abgesetzte Tastatur nichts nützt, wenn der Reset fast nur durch Aufstehen und Umrunden des Schreibtischs auszulösen ist. Und wer jemals ein Programm auf dem ATARI ST entwickelt hat, der weiß, wie oft so ein Reset zwangsweise ausgelöst werden muß, weil der Rechner sich mal wieder haltlos in seinen Innereien verstrickt hat.

Um dem jetzt Abhilfe zu schaffen, bieten wir allen Interessenten in der Programmierpraxis eine Software-Lösung – quasi einen verlängerten Arm zur Reset-Taste – an, die durch gleichzeitiges Drücken der beiden Shift-Tasten und der Control-Taste einen Software-Reset beim ST auslöst. Das Assembler-Programm ist auf allen ST's lauffähig, d.h., man kann es sowohl mit dem BlitterTOS als auch mit der letzten TOS-Version vom 6.2.1986 benutzen. Die Routine ist gut dokumentiert, so daß man mit dem Verständnis keinerlei Probleme haben dürfte. Wir wollen nur zwei Punkte aus dem Listing herausgreifen, die im install-Teil zu finden sind. Da wäre zuerst das Laden des VBL-Slots (VBL = Vertical Blank = Bildrücklauf). Die VBL-Queue des TOS wird ständig interruptmäßig abgefragt, so daß man hier leicht Zeiger auf eigene Routinen, die durch Tastatureingabe ausgelöst werden, einbinden kann. Standardmäßig stehen acht VBL-Slots zur Verfügung. Es können aber weitere initialisiert werden. Für unsere Routine ist es unbedingt wichtig, den zweiten Slot zu nehmen. Das hat folgende Gründe: Normalerweise wird man eine Reset-Routine im Auto-Ordner installieren, so daß sie beim Einschalten des Rechners automatisch geladen und gestartet wird. Nach dem Abarbeiten des Auto-Ordners wird aber erst das GEM initialisiert und dadurch der erste Slot überschrieben. Folglich würde in diesem Falle auch der Zeiger auf unsere Reset-Routine gelöscht.

Der zweite Punkt betrifft die Adresse von kb–shift in den verschiedenen TOS-Versionen. Diese Adresse wird dazu benötigt, die Tastatureingaben (in unserem Falle Shift + Shift + Control) abzufragen. Da nun je nach TOS-Version (Blitter-TOS oder altes TOS) der Zeiger auf kb–shift an einer anderen Stelle im Speicher zu suchen ist, müssen beide Möglichkeiten abgefragt werden. Im alten TOS war die Adresse leider nicht dokumentiert, so daß wir in diesem Falle den nicht ganz legalen direkten Zugriff darauf wagen müssen.

(HE)

Listing :

```
 1:  *      Reset auf Shift + Shift + Control
 2:  *      Oktober 1987 (C) Oliver Joppich
 3:
 4:  .globl     _main
 5:
 6:
 7:  _main      movea.l 4(a7),a0          : Adresse der Basepage
 8:             move.l  $C(a0),d0         : Code Länge
 9:             add.l   $14(a0),d0        : plus Data Länge
10:             add.l   $1C(a0),d0        : plus bss Länge
11:             add.l   #$100,d0          : plus Größe der Basepage
12:             move.l  d0,save_len       : effektive Programmlänge
                                            sichern
13:             clr.l   -(sp)             : Parameter 0L
14:             move.w  #$20,-(sp)        : Supervisor ein
15:             trap    #1                : jetzt
16:             addq.l  #6,sp             : Stack korrigieren
17:             move.l  d0,save_stack
18:
19:             jsr     install           : VBL installieren
20:
21:             move.l  save_stack,-(sp)  : alten Stack holen
22:             move.w  #$20,-(sp)        : wieder in den User
                                            Modus
23:             trap    #1                : jetzt
24:             addq.l  #6,sp             : Stack korrigieren
25:
26:             move.w  #$0,-(sp)         : Parameter 0
27:             move.l  save_len,-(sp)    : Länge des Programms
28:             move.w  #$31,-(sp)        : Stay and keep resident
29:             trap    #1                : jetzt
30:
31:  install    move.l  $456,a0           : VBL Slot laden
32:             move.l  4(a0),oldvbl      : 2. nehmen und retten
                                            (wegen AUTO-Ordner)
33:             lea     job,a1            : neuen VBL laden
34:             move.l  a1,4(a0)          : und installieren
35:             move.l  ($fc0024),kb_p    : Adresse von Kbshift ab
                                            Blitter TOS
36:             cmpi.b  #$86,$fc001b      : altes TOS von 86 ?
37:             bne     new_tos           : nein !
38:             move.l  #$e1b,kb_p        : nicht legalen Zugriff
                                            wagen
39:  new_tos    rts
40:
41:  job        move.l  (kb_p),a0         : Vektor auf Kbshft
42:             cmpi.b  #$7,(a0)          : Treffer auf s+s+c ?
43:             bne     end_vbl           : nö, also weiter
44:             move.l  $fc0004,a0        : Reset Adresse laden
45:             jmp     (a0)              : und jupp ab ins Nirwana
46:
47:  end_vbl    movea.l oldvbl,a0         : jetzt den anderen VBL
48:             cmpa.l  #$0,a0            : war einer da ?
49:             beq     nomain            : nö, also zurück
50:             jsr     (a0)              : jetzt der andere VBL
51:  nomain     rts
52:
53:  save_len          ds.l 1             : Zwischenspeicher für
                                            PRG Länge
54:  kb_p              ds.l 1             : Vektor auf Kbshift
55:  oldvbl            ds.l 1             : Vorgänger VBL in SLOT 2
56:  save_stack        ds.l 1             : Stackzwischenspeicher
```



GFA-Basic ruft Maschinencode, bitte kommen...

...Verstanden. Antworten...

Wenn es nur so einfach gehen würde, dachte ich mir schon oft, als ich die beiden GFA-Basic-Befehle CALL und C: als Schnittstelle für kleine Maschinenprogramme benutzen wollte. Doch die vorhandene Literatur (Benutzer-Handbuch, GFA-Basic-Buch von F. Ostrowski, Data Becker's GFA-Buch, diverse Zeitschriften) schweigen sich beharrlich über dieses Thema aus, vorallem was den Befehl CALL betrifft.

Der parameterlose Aufruf mittels CALL bietet ja keine Probleme, also rufe ich meine Prozedur so auf, nachdem ich alle Zahlen und Strings mühsam gepoked habe. Nun, dies mag wohl für kleine Testproceduren hinhalten, doch auf die Dauer ist sowas kein Programmierstil und dazu noch kaum lesbar!

Jetzt aber zur Sache! Als erstes widme ich mich dem C:-Aufruf, da dieser einfacher zu handhaben ist. Er ist weniger komplex als der CALL-Aufruf, dafür können jedoch keine Strings übergeben werden.

Mit dem C:-Aufruf können (keiner oder mehrere) Parameter übergeben werden. Als Parametertypen sind erlaubt: Integer-, Fließkomma- und bool'sche Zahlen als Variablen sowie auch als Direktwerte (siehe Bild 1). Die Parameter werden standardmäßig in 16-Bit-Integer-Zahlen konvertiert und auf den Stapel geschoben. Wenn vor einem Parameter der Zusatz L: steht, wird der 32-Bit Wert berechnet und auf dem Stapel in der üblichen Weise abgelegt. Der erste Wert (jener ganz links in der Parameterliste) liegt auf dem Stapel ab 4(SP), der zweite ab 6(SP) wenn der erste Wortlänge hatte, sonst ab 8(SP) usw. Bei (SP), also an der Spitze des Stapels, liegt die Langwortadresse für den Rücksprung, denn die Maschinenroutine wird intern mit JSR (A0) aufgerufen. Doch dies betrifft uns bei der Programmierung nur insofern, als diese Adresse nicht verändert werden darf und der Stapelzeiger den gleichen Wert hat wie beim Aufruf der Routine (das Gleiche gilt für CALL). Somit dürfte auch klar sein, daß das ganze Programm mit RTS (&H4E75) abgeschlossen werden muß.

Ist die Routine abgearbeitet, wird der im Datenregister 0 gespeicherte Langwort-Wert ans Basic zurückgegeben. Auf diese Weise kann – wie bei einem Funktionsaufruf – ein Wert an eine Variable übergeben werden oder, falls nicht benötigt, mittels der Pseudo-Variablen Void ignoriert werden.

Sollte eine Problemstellung mehrere Rückgabeparameter erfordern, so kann mit dieser Möglichkeit in D0 die Adresse eines Speicherblockes übergeben werden, wo sich die entsprechenden Daten befinden.

Werden keine Parameter von der Prozedur benötigt, muß trotzdem eine leere Parameterliste da sein, (z.B. Void C:Versuch()), im Gegensatz zu Call, das dann so aussähe: Call Versuch

Ein Tip noch: Hat die Prozedure Parameter, so ist am Anfang der erste Wert in D0 abgelegt. Der Aufruf Print C:Test(123) hätte also, würde das Programm nur aus RTS bestehen, die Ausgabe "123" zur Folge.

Damit wäre mal die erste Möglichkeit ein wenig ausgeleuchtet. Diese Funktion können Sie noch am Beispiel der Laufschrift näher studieren.

Nun zum CALL-Aufruf. Als erstes sei mal gesagt, daß CALL in der aktuell vorliegenden GFA-Basic Version 2 nicht fehlerfrei implementiert ist und deshalb nur bedingt verwendbar ist. So kommt es bei mir z.B. regelmäßig zu einem Absturz, wenn ich CALL mit einem direkten String ("Hallo") als Parameter aufrufe und danach in den Direktmodus schalte (mit der ESC-Taste). Die Maus reagiert noch, doch hängt sich scheinbar der Rest der Tastatur auf. Wird aber nicht in den Direktmodus gegangen, oder erst, nachdem das Programm mit NEW gelöscht worden ist, ist nichts zu befürchten. Auch bei Aufrufen mit numerischen Parametern sind bisher noch keine Störungen aufgetreten.

Trotzdem ist es mir nach eini-

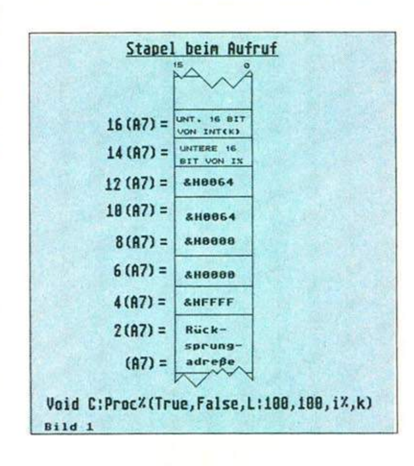


Bild 1
Bei dem C-Aufruf sind folgende Parametertypen erlaubt: Integer-Fließkomma- und bool'sche Zahlen als Variablen sowie auch als Direktwerte.

Die Daten werden z.B. folgendermaßen vom Stapel geholt: (z.B. Ret%=C:Start(Anzahl%,L:Adresse%,Offset%))

```
START:    MOVE.W    4(SP),D1    * = Anzahl%
          MOVE.L    6(SP),A0    * = Adresse%
          ADDA.L    A(SP),A0    * = Offset%
          .
          .
          .
          MOVE.L    A0,D0       * Ret% := A0
          RTS
```

gen Anläufen gelungen, das Geheimnis dieses Befehles zu lüften.

Der Call-Befehl ist nicht mit einer Funktion vergleichbar, sondern er führt das Programm ohne Echo aus. Die Parameterübergabe hingegen ist universeller als bei C:. Es sind grundsätzlich alle eindimensionalen Parameter erlaubt, also zusätzlich zu jenen von C: auch noch Strings (als Variable sowie als Direktwert). Es muß jedoch jedes Mal ein sogenannter Dummywert an die Parameterliste gehängt werden, da der letzte Wert dieser Liste vom Interpreter listigerweise einfach ignoriert wird. Dafür wird meines Erachtens völlig unnötig als erster Wert die Adresse der Prozedur noch in der Werteliste im Speicher abgelegt. Diese beiden Verhaltensweisen des Interpreters sind wahrscheinlich auf die falsche Programmierung eines Laufindexes zurückzuführen. Gut, diesen Fehler können wir nun relativ einfach umgehen mit der besagten "Dummy-Strategie".

Beim Aufruf sieht der Stapel wie folgt aus: Zuerst kommt die Langwortrücksprungadresse, darauf folgt bei 4(SP) ein Wort, welches die Anzahl Parameter (inclusiv Dummy-Wert) angibt, und zu guter Letzt folgt bei 6(SP) die Langwortadresse, die auf einen Speicherbereich zeigt, wo die Werte abgelegt sind. Dort steht zuerst die Adresse der Routine. Darauf folgen die Parameter. Jeder Parameter belegt vier Byte im Block (außer dem Dummy, der sich mit NULL BYTE begnügt). Die numerischen Parameter werden analog zu C: in Integerzahlen konvertiert, nur geschieht das hier mit allen Werten auf einer Breite von 32 Bit (siehe Bild 2).

Für die Strings ist in ihren vier Bytes die Adresse abgelegt, ab welcher der String zu finden ist. Doch hier beginnt nun der eigentliche Haken der ganzen Sache. Ist die Länge des Strings gerade, folgt auf das letzte Zeichen ein Null-Byte (auf einer geraden Adresse). Ist die Länge jedoch ungerade, folgt auf das letzte Zeichen ein zufälliger Wert und erst darauf das Null-Byte an der geraden Adresse. Also kann die wirkliche Länge des Strings beim besten Willen daraus nicht ersehen werden. Für Strings, die in Form einer Variablen übergeben werden, kann die Länge auf andere Art bestimmt werden. Auf das Null-Byte folgt eine Drei-Byte-Adresse (mit dem führenden Null-Byte zusammen ergibt das eine Langwortadresse, doch die oberen acht Bit sind nicht nötig für die Adressierung, da der Adressbus sowieso nur 24 Bit breit ist). Diese Adresse zeigt auf einen sechs Byte umfassenden Block. Sie ist dieselbe, die bei der Basic-Funktion Arrptr(String$) zurückgegeben wird. Die ersten vier Bytes zeigen wiederum auf den String, was der Funktion Varptr(String$) entspricht. Die nächsten beiden Bytes geben die Länge des Strings wieder (siehe Bild 3).

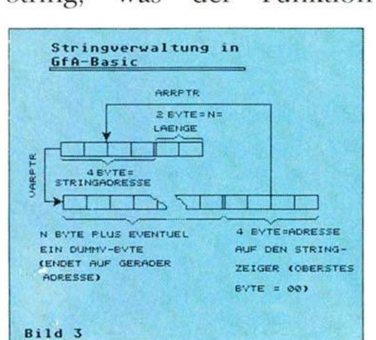


Bild 3
Die ersten vier Bytes zeigen auf den Stringzeiger. Die nächsten beiden Bytes geben die Länge des Strings wieder.

Die Stringeingabe kann man aber trotzdem auf verschiedene Weise absturzsicher implementieren.

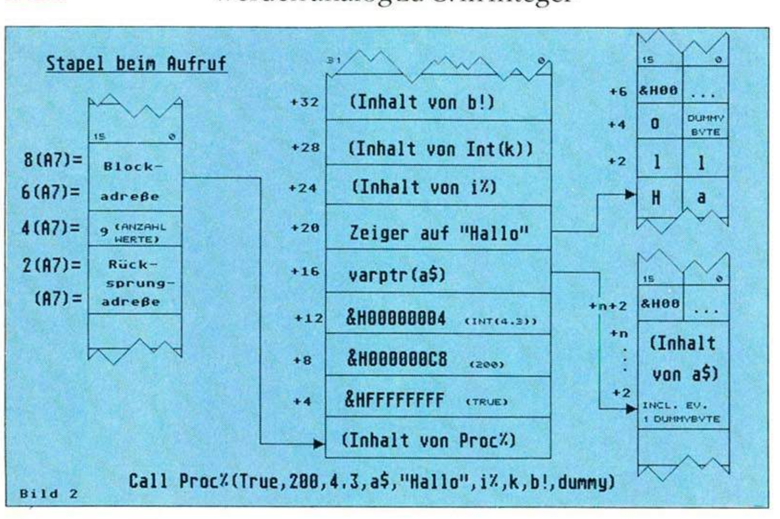
Es könnte z.B. vor jedem String die Länge übergeben werden, oder die Strings dürfen nicht als Direktwert angegeben werden. Ein anderer Weg, der auch mit C: begangen werden könnte, ist die Übergabe von Arrptr(String$). Damit ist die Adresse des Strings und die Länge bekannt.

Eigentlich schade, daß ein so flexibler Befehl wie der Call-Befehl im Sumpf der Programmierfehler verelendet. Scheinbar hat noch kaum jemand ernsthaft mit diesem Befehl programmiert, sonst wäre wohl auch mehr darüber in der Literatur nachzulesen gewesen. Wer weiß, ob Frank Ostrowski von diesen fehlerhaften Befehl schon etwas vernommen hat. Das soll kein Vorwurf an den Programmierer dieses sonst brillianten und höchst zuverlässigen Basics sein! Der Call-Befehl ist vielleicht auch in Vergessenheit geraten, weil die Alternative im C:-Befehl existiert, und weil dieser einfacher zu handhaben ist und vorallem auch noch die Möglichkeit der Parameterübergabe ans Basic zurück vorsieht.

Ein Nachtrag noch zur Konvertierung der Fließkommazahlen in Integer:

Die Funktion Int(Wert) entspricht nicht der Basic-Funktion Int! Bei der Basic-Funktion wird aus -4.3-5 = Int(-4.3). Bei der Funktion der Konversion wird daraus -4 = Int(-4.3).

Bild 2
Die numerischen Parameter werden analog zum C-Aufruf in Integerzahlen konvertiert. Der Unterschied liegt darin, daß hier alle Werte auf einer Breite von 32 Bit umgewandelt werden.



Basic-Erweiterung selbst gestrickt

Nun will ich als kleine Demonstration der beiden Befehle eine kleine Basic-Erweiterung anregen, damit jederman seine eigenen Ideen verwirklichen kann.

Beim Beispiel handelt es sich um die Programmierung einer Rollschrift. Das Programm ist in zwei Teile gegliedert: in eine Initialisierung (wo die Koordinaten der zu rollenden Box gespeichert werden) und in den eigentlichen Roll-Befehl (wo die Anzahl zu rollender Bits und ein Synchronisationswert, der es erlaubt, die Verzerrung auf ein Minimum zu reduzieren, eingegeben werden). Die Funktion läuft nur auf Monochrom Monitor, da direkt auf den Bildschirm-Ram zugegriffen wird. In der Prozedur wurde bewußt auf 'Vsync' verzichtet, da dies erstens eine enorme Verzögerung mit sich bringt und zweitens je nach Boxgröße unschöne Verzerrungen gibt. Stattdessen kann mit einem Zahlenwert nach jedem Bitschritt eine entsprechende Warteschleife durchlaufen werden, so daß die Rotation mit dem Strahlrücklauf synchronisiert werden kann. Dadurch wird die Verzerrung zwar nicht verschwinden, doch sie kann damit in einen Bereich 'verschoben' werden, wo sie nicht stört ober garnicht sichtbar ist. Das Demo-Programm (Listing 3) zeigt, wie der Abgleich ablaufen könnte. Aber wie immer: experimentieren ist erlaubt!

Die Initialisierung geht folgendermaßen:

Bit% = C:Scrollinit%(X,Y,X1,Y1)

Wobei X/Y die obere linke, X1/Y1 die untere rechte Ecke ist. Die Werte X und X1 werden auf Vielfache von 16 gerundet, daraus wird die Anzahl der Bits pro Zeile berechnet und nach Bit% übergeben (siehe Bild 6). Die Parameter werden der Einfachheit halber nicht auf den gültigen Bereich hin überprüft (siehe Bild 4).

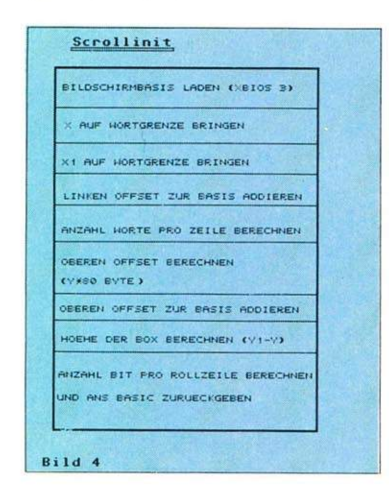


Bild 4 Ein sauberer Überblick über das gesamte 'Scrollinit' Procedure.

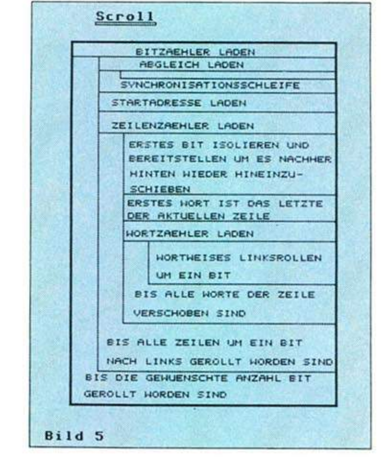
Ein Bereichskonflikt ist im allgemeinen mit einem Absturz verbunden!

Der wiederkehrende Aufruf lautet: (siehe Bild 5)

Call Scroll%(Anzahl%,Abgleich%,Dummy)

Anzahl% gibt die Anzahl Bit an, die mit diesem Aufruf gerollt werden sollen, und Abgleich% ist der Synchronisationswert (positiv!).

Bild 5 Die Procedure, die das Bild 'laufen' läßt, ist hier zu sehen.



Die Berechnung der Box

X AUF WORTADRESSE GEBRACHT

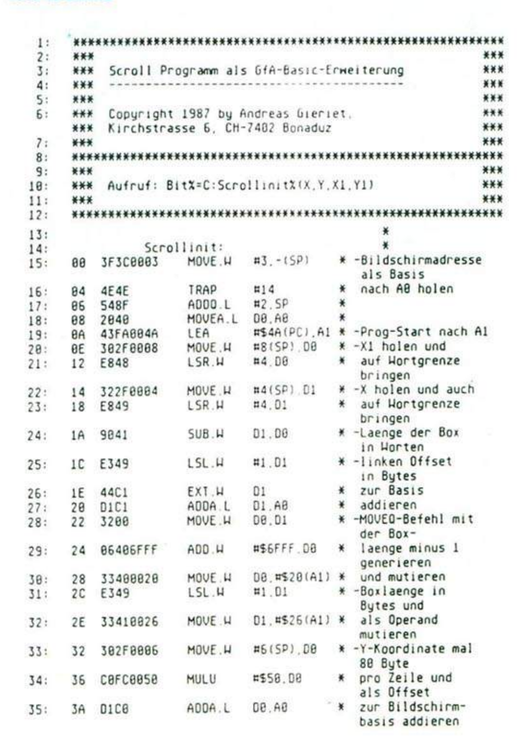
X1 AUF WORTADRESSE GEBRACHT

Y X Y1 X1

Bild 6

Bild 6 Die Berechnung der Box, deren Inhalt 'Scrolling' sein soll, wird sehr raffiniert gemacht.

Listing 1: Die Assembler-Routine



```
 1:      *************************************************************
 2:      ***                                                       ***
 3:      ***  Scroll Programm als GfA-Basic-Erweiterung            ***
 4:      ***  ----------------------------------------             ***
 5:      ***                                                       ***
 6:      ***  Copyright 1987 by Andreas Gieriet,                   ***
         ***  Kirchstrasse 6, CH-7402 Bonaduz                      ***
 7:      ***                                                       ***
 8:      *************************************************************
 9:      ***                                                       ***
10:      ***  Aufruf: Bit%=C:Scrollinit%(X,Y,X1,Y1)                ***
11:      ***                                                       ***
12:      *************************************************************
13:                                            *
14:               Scrollinit:                  *
15:  00  3F3C0003   MOVE.W   #3,-(SP)       * -Bildschirmadresse
                                               als Basis
16:  04  4E4E       TRAP     #14            *  nach A0 holen
17:  06  548F       ADDQ.L   #2,SP          *
18:  08  2040       MOVEA.L  D0,A0          *
19:  0A  43FA004A   LEA      #$4A(PC),A1    * -Prog-Start nach A1
20:  0E  302F0008   MOVE.W   #8(SP),D0      * -X1 holen und
21:  12  E848       LSR.W    #4,D0          *  auf Wortgrenze
                                               bringen
22:  14  322F0004   MOVE.W   #4(SP),D1      * -X holen und auch
23:  18  E849       LSR.W    #4,D1          *  auf Wortgrenze
                                               bringen
24:  1A  9041       SUB.W    D1,D0          * -Laenge der Box
                                               in Worten
25:  1C  E349       LSL.W    #1,D1          * -linken Offset
                                               in Bytes
26:  1E  44C1       EXT.W    D1             *  zur Basis
27:  20  D1C1       ADDA.L   D1,A0          *  addieren
28:  22  3200       MOVE.W   D0,D1          * -MOVEQ-Befehl mit
                                               der Box-
29:  24  06406FFF   ADD.W    #$6FFF,D0      *  laenge minus 1
                                               generieren
30:  28  33400020   MOVE.W   D0,#$20(A1)    *  und mutieren
31:  2C  E349       LSL.W    #1,D1          * -Boxlaenge in
                                               Bytes und
32:  2E  33410026   MOVE.W   D1,#$26(A1)    *  als Operand
                                               mutieren
33:  32  302F0006   MOVE.W   #6(SP),D0      * -Y-Koordinate mal
                                               80 Byte
34:  36  C0FC0050   MULU     #$50,D0        *  pro Zeile und
                                               als Offset
35:  3A  D1C0       ADDA.L   D0,A0          *  zur Bildschirm-
                                               basis addieren
```

```
36:  3C  23480014   MOVE.L   A0,#$14(A1)    * -als Operand
                                               mutieren
37:  40  302F000A   MOVE.W   #$A(SP),D0     * -Hoehe der Box (Y1-Y)
38:  44  906F0006   SUB.W    #6(SP),D0      *  berechnen und
                                               als Zeilen-
39:  48  5340       SUBQ.W   #1,D0          *  zaehler (korrekt
                                               minus 1)
40:  4A  3340001A   MOVE.W   D0,#$1A(A1)    *  als Operand
                                               mutieren
41:  4E  3001       MOVE.W   D1,D0          * -Boxlaenge von
                                               Bytes
42:  50  E748       LSL.W    #3,D0          *  in Bits umrechnen
                                               und
43:  52  44C0       EXT.W    D0             *  als Rueckgabewert
                                               speichern
44:  54  4E75       RTS                     *
45:                                         *
46:      *************************************************************
47:      ***                                                       ***
48:      *** Aufruf: Call Scroll%(Bit%,Abgleich%,0)                ***
49:      ***                                                       ***
50:      *************************************************************
51:                                                *
52:                     Scroll:                    *
53:  56  226F0006           MOVEA.L  #6(SP),A1     * -Anzahl zu ver-
                                                     schiebende Bit
54:  5A  24290004           MOVE.L   #4(A1),D2     *  als korrekten
                                                     Schleifen-
55:  5E  5382               SUBQ.L   #1,D2         *  zaehler
                                                     speichern
56:  60  26290008       L2: MOVE.L   #8(A1),D3     * -Abgleich-
                                                     Zaehler
57:  64  51CBFFFE       L1: DBF      D3,L1         * -Sync-Schleife
58:  68  207C00078000       MOVEA.L  #$78000,A0    * -Adresse von
                                                     X/Y (mutiert)
59:  6E  323C000A           MOVE.W   #$10,D1       * -Zeilenzaehler
                                                     (mutiert)
60:  72  3010           L3: MOVE.W   (A0),D0       * -erstes Bit
                                                     isolieren um
                                                     es
61:  74  E350               ROXL     D0            *  hinten hinein-
                                                     zuschieben
62:  76  7027               MOVEQ    $#27,D0       * -Wortzaehler
                                                     (mutiert)
63:  78  D1FC00000050       ADDA.L   $#50,A0       * -bei X1 begin-
                                                     nen (mutiert)
64:  7E  E5E0           L4: ROXL     -(A0)         * -von hinten
                                                     nach links
                                                     rollen
65:  80  51C8FFFC           DBF      D0,L4         * -bis die ganze
                                                     Zeile fertig
66:  84  D1FC00000050       ADDA.L   $#50,A0       * -Adresse der
                                                     naechsten
                                                     Zeile
67:  8A  51C9FFE6           DBF      D1,L3         * -bis alle
                                                     Zeilen
68:  8E  51CAFFD0           DBF      D2,L2         * -bis alle Bit
69:  92  4E75               RTS                    * -ENDE
```

Listing 2: Diese Routine schreibt das Scroll-Programm auf Diskette

```
1:   Open "O",#1,"SCROLL.PRO"                ! Name der Datei
2:   Clr Test%                               ! Check-Summe
3:   Do                                      ! Einleseschleife
4:     Read A$
5:     Exit If A$="***"                      ! Abbruchbedingung
6:     A%=Val("&H"+A$)
7:     Print #1,Mki$(A%);                    ! als 2-Byte-Wert
                                               speichern
8:     Add Test%,A%                          ! Summieren fuer Test
9:   Loop
10:  Close #1                                ! korrekt schliessen
11:  Read A$                                 ! Check-Summe lesen
12:  If Val("&H"+A$)<>Test%                  ! vergleichen und
                                               reagieren
13:    Kill "SCROLL.PRO"
14:    Print "  Ein Datenfehler ist aufgetreten! (Datei
         geloescht)"
15:  Else
16:    Print "  Alle Daten sind korrekt unter 'SCROLL.PRO'
         gespeichert!"
17:  Endif
18:  End
19:  '
20:  Daten:                                  ! Daten
21:  Data 3F3C,0003,4E4E,548F,2040,43FA,004A,302F
22:  Data 0008,E848,322F,0004,E849,9041,E349,44C1
23:  Data D1C1,3200,0640,6FFF,3340,0020,E349,3341
24:  Data 0026,302F,0006,C0FC,0050,D1C0,2348,0014
25:  Data 302F,000A,906F,0006,5340,3340,001A,3001
26:  Data E748,44C0,4E75,226F,0006,2429,0004,5382
27:  Data 2629,0008,51CB,FFFE,207C,0007,8000,323C
28:  Data 000A,3010,E350,7027,D1FC,0000,0050,E5E0
29:  Data 51C8,FFFC,D1FC,0000,0050,51C9,FFE6,51CA
30:  Data FFD0,4E75
31:  Data ***
32:  Data 001804C1
```

Listing 3: Demonstrationsprogramm

```
1:   Dim Scroll%(36)                         ! Platz fuer das Programm
2:   Scrollinit%=Varptr(Scroll%(0))          ! Start des Initialisie-
                                               rungsprogrammes
3:   Scroll%=Scrollinit%+86                  ! und des Scroll-Program-
                                               mes
4:   Bload "SCROLL.PRO",Scrollinit%          ! Das Programm laden
5:   '
6:   Pbox 29,10,610,38                       ! Kleine Testgrafik zeich-
                                               nen
7:   Deffill 0
8:   Pbox 32,13,607,35
9:   Deffill 1
10:  Print At(11,2);Chr$(189);" 1987 by  Andreas Gieriet,
       Kirchstrasse 6, CH-7402 Bonaduz"
11:  '
12:  Laenge%=C:Scrollinit%(32,18,608,32)  ! Initialisierungs-
                                            aufruf
13:  '
14:  Hidem                                  ! Maus verschwindet,
                                              damit es keinen
15:  Abgleich%=0                            ! Kollision gibt
16:  Print At(15,10);"Abgleich mit :"
17:  Do
18:    Print At(33,10);Abgleich%
19:    Vsync                                ! Nicht noetig
20:    Call Scroll%(Laenge%,Abgleich%,0)    ! Mit verschiedenen
                                              Abgleichwerten
21:    Pause 20                             ! eine ganze Breite
                                              rollen
22:    Add Abgleich%,100                    ! Naechster Abgleich-
                                              wert
23:  Loop                                   ! Abbruch nur mit
                                              CONTROL/SHIFT/
                                              ALTERNATE
24:  Showm                                  ! Maus wieder
                                              einschalten
25:  End
```

Nun wünsche ich allen viel Spaß beim Erweitern des GFA-Basics und vielleicht hören wir auch mal von Ihnen etwas über eine 'selbst gestrickte' Erweiterung!

Andreas Gieriet

Die Evolution des Weltalls in APL oder: Punkte auf dem Würgel

Haben Sie schon mal wissen wollen, wie unser Weltall entstanden ist?

Oder interessiert es Sie mehr, wie es sich weiterentwickeln wird?

Diese Fragen werden hier nicht beantwortet. Das sollte Sie nicht enttäuschen; immerhin käme es einer mittleren Sensation gleich, wenn die Fragen, mit denen sich die Menschheit befaßt, seit sie denken kann, und die immer noch offen sind, in dieser Ausgabe der ST Computer gelöst würden.

Sie sehen hier aber ein kleines Programm, das einen Teil-Aspekt der Welten-Problematik beantwortet: nämlich die Frage, ob die Entwicklung eines Systems, die über lange Zeiträume kontinuierlich erscheint, immer weiter fortschreiten muß oder ob sie nicht vielleicht doch mal aufhört.

Das Modell ...

Mit solchen Hintergedanken im Kopf habe ich mir ein kleines, (sehr, sehr kleines) System ausgedacht, um damit eine Evolution zu simulieren:

Ich konstruiere ein Feld, dessen Größe prinzipiell gleichgültig ist; ich schlage 10 Zeilen und 39 Spalten vor. Dieses Feld hat noch eine Besonderheit: Es hat keine Grenzen! D.h., die oberste zeile schließt an die unterste an und die erste Spalte an die letzte. Wem jetzt Wortschöpfungen wie "Kufel" oder "Würgel" durch den Kopf gehen, liegt nicht verkehrt: Wir haben eine eckige Kugel oder einen runden Würfel vor uns.

Die Geschichte beginnt damit, daß ich in die Mitte des Feldes (aus der Sicht des Betrachters) zwei Punkte oder Einsen setze, die ich in meinem Modell zu sozialen Wesen erkläre. Ich behaupte, daß einerseits (zumindest in einer Welt, wie wir sie kennen) ein Einzelwesen kaum überleben kann, daß aber andererseits die Überlebenschancen bei Überfüllung der Welt sinken.

Um die Sache mathematisch zu vereinfachen und zu präzisieren, stelle ich also für die Weiterentwicklung meiner Punkte folgende Bedingung auf:

In jedem neuen Zeitabschnitt (Generation, Durchlauf) sollen sich nur auf den Feld-Positionen Punkte befinden, die im Zeitabschnitt zuvor genau zwei direkte Nachbarn hatten (in allen Richtungen, also auch diagonal).

... in APL

Die Umsetzung dieses Modells in APL erfordert ein bißchen Knobelei, ist aber letzten Endes ganz einfach: Die eigentliche Rechnerei nimmt eine ganze Zeile in Anspruch (ich gebe zu: eine lange). Alles andere ist das Drumherum und die Bildschirmanzeige.

Zwei Einschränkungen habe ich zu machen, weil ich das Wesentliche an diesem Programm herausstellen wollte:

- Ich behaupte nicht, daß mein Weg der schnellste ist.
- Die Bildschirmanzeige ist nicht graphisch aufgemotzt.

Um die Anzahl der Nachbarschaften für jede einzelne Feld-Positionen zu ermitteln, rotiere ich die ganze Matrix um jeweils ein Element nach oben, unten, links und rechts: Das kostet mich in APL schließlich nur ein Lächeln; und weil's so schön war, darf ich gleich noch mal lächeln: Da ja auch die diagonalen Nachbarn berücksichtigt werden sollen, nehme ich jeweils die nach oben und unten verschobene Matrix und rotiere diese auch noch mal nach rechts und links.

Und wozu das Ganze? Ich brauche jetzt nur noch, bildlich gesprochen, diese ganzen rotierenden Matrizen (immerhin acht Stück) aufeinander zu legen und zu schauen, wo genau zwei Einsen übereinander liegen! Oder anders ausgedrückt: Ich vergleiche 2 mit der Summe dieser Matrizen. Das Addieren und Vergleichen von Matrizen ist in APL ja auch kein Problem.

Sie sehen schon: Die Erklärung dieser einen logischen Zeile im Programm nimmt viele Zeilen im Text in Anspruch. Viel einfacher ist es, das Ganze einfach mal auszuprobieren.

APL ist eben kürzer.

Die beiden abgebildeten Funktionen tun übrigens genau das gleiche, die zweite ist nur zur Demonstration auf Kürze getrimmt:

- die 1. Zeile schafft die Voraussetzungen,
- die 2. bringt das Ergebnis auf den Bildschirm,
- die 3. rechnet.

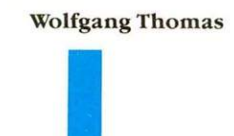
So, nun werden Wetten angenommen: Was wird passieren? Geht die Entwicklung weiter, bis jemand CTRL-C drückt oder den Netzstecker zieht? Sterben die Punkte aus? Wird das Feld wegen Überfüllung geschlossen? Oder gibt es sogar noch eine weitere Möglichkeit?

Ich setze darauf, daß Sie, wenn Sie sich etwa 60 Minuten Zeit nehmen, spätestens beim 1767. Durchlauf überrascht sind!

```
Desk  File  Edit  Break  Options
APL.68000
EVOLUTION1;L;M1;M2;MX;R
⍝ --------- Startbedingungen ------------------------------------
R←⎕CC 1 ⍝                     Bildschirm leeren
MX←10 39⍴0 ⍝                  Matrix mit 10 Zeilen und 39 Spalten Nullen
MX[5 6;20]←1 ⍝                ... und zwei Einsen (z.B.)
L←0 ⍝                         Anzahl der Durchläufe
⍝ --------- Anzeige ---------------------------------------------
L1:R←⎕CC 0 0 0 ⍝              Cursor in die linke obere Ecke
39↑(⍕+/,MX),' Punkte / ',(⍕L←L+1),'. Lauf',25⍴'_' ⍝ Anzeige: Zahl, Lauf
10 39⍴(,MX)\'*' ⍝              Anzeige: Einsen als Punkte
⍝ --------- Rechnung --------------------------------------------
MX←2=(1⌽M2)+(¯1⌽M2)+(1⌽M1)+(¯1⌽M1)+(1⌽MX)+(¯1⌽MX)+(M2←1⊖MX)+(M1←¯1⊖MX)
⍝ -------------------------------------------------------------
→L1 ⍝                         Loop
```

Abbildung 1: Das Programm in APL

Wolfgang Thomas



60 Hz-Umschaltung in der Bootspur

Besitzern von Farbmonitoren, die diese an einem ATARI ST betreiben, fällt meist zuerst eine – vor allem gemessen an dem guten Bild des SW-Monitors SM124 – unangenehme Flimmereigenschaft auf. Dies liegt daran, daß der Farbmonitor an den PAL-STs standardmäßig im 50-Hz-Modus betrieben wird, obwohl fast alle normalen Monitore auch 60 Hz verkraften. Dieser Artikel zeigt eine Möglichkeit, auch ohne Betriebssystemänderung praktisch jedes Programm auch mit 60 Hz laufen zu lassen, und das ohne zusätzlichen Aufwand beim Start der Diskette.

Das hier vorgestellte Programm läuft unter GFA-Basic und läßt sich ohne Probleme compilieren. Es ist von der TOS-Version unabhängig und läuft (bei mir) mit jeder Speicherkonfiguration.

Wichtige Vorbemerkung:

Obwohl bisher noch absolut keine Schwierigkeiten aufgetreten sind, sollten grundsätzlich nur Sicherheitskopien, keine Originale, mit dem Programm bearbeitet werden – man weiß ja nie, ob irgendein Kopierschutz nicht gerade die veränderten Diskettenbytes abfragt.

Bootsektoren

Hier noch einmal die genaue Definition der Aufgabe: Nach einem Reset soll die Speicherstelle $ff8201, die die 50/60-Hz-Ausgabe des Videochips steuert, gelöscht werden, und das möglichst ohne zusätzlichen Aufwand und unter Verwendung von möglichst wenig Speicherplatz auf der Diskette.

Aus naheliegenden Gründen kann ein Programm im Auto-Ordner hier nicht weiterhelfen, da bei kommerziellen (insbesonders Spiel-) Programmen die normale Diskettenstruktur oft nicht eingehalten wird (oder ganz einfach kein Platz mehr auf der Diskette ist). Außerdem muß ein Auto-Ordner-Programm immer erst gesucht und geladen werden.

Es bietet sich daher an, bei Disketten, die auch zusammen mit dem Farbmonitor benutzt werden, einen Bootsektor zu installieren, der bei einem Reset automatisch ausgeführt wird und auf 60 Hz umstellt. Bootsektor einer Diskette ist immer der erste Sektor, der sich auf ihr befindet. Beträgt die (Wort-) Summe aller Wörter in ihm $1234, so wird angenommen, daß sich ein ausführbares, im Speicher frei verschiebbares Installationsprogramm auf ihm befindet (ab Sektoradresse 0), und er wird im Verlauf der Reset-Routine (früher als z.B. der Auto-Ordner) ausgeführt (Rückkehr in Betriebssystem mit 'RTS'). Da sich dort aber auch noch andere Informationen befinden, hier ein kurzer schematischer Aufbau:

$00-$01
BRA.S anf ;Branch zum Anfang des Programms
$02-$07
OEM-Bereich, ungenutzt (?)
$08-$1D
Hier wird die Anzahl der Tracks usw. definiert
$1E-$1FD
frei für Installationsprogramme
$1FE-$1FF
wird so berechnet, daß die Quersumme $1234 beträgt

Interessant ist hier: Es befinden sich am Anfang des Bootblocks zwei freie Bytes, die bei Bootsektoren einen Sprung zum Anfang des eigentlichen Installationsprogramms bewirken. Normalerweise wird hier der relative Sprung mit BRA.S benutzt. Es schließen sich 6 Bytes an, die als OEM-Bereich bezeichnet und nicht benutzt werden. Da das Installationsprogramm, das hier in die Diskette eingeschleust werden soll, äußerst kurz ist (lösche $ff8201, also clr.b $ff8201, und springe dann zurück), und möglichst Kompatibilität mit Disketten gewahrt bleiben soll, die den Bootblock bereits selbst benutzen, geht das Programm so vor:

Ausführung:

1. Lies den Bootsektor der augenblicklich eingelegten Diskette ein.

2. Summiere alle Wörter des Bootblocks. Wenn die Summe $1234 beträgt, notiere, daß der Bootblock bereits ausführbar ist, und frage, ob der Bootblock überschrieben werden oder ob das alte Installationsprogramm nach dem neuen ausgeführt werden soll. Wenn Letzte-

Listing

```
 1:    '    Makeparm V2.1
 2:    '    Ändert den Parameterblock, so daß beim Start von Disk
 3:    '    auf 60 Hz umgeschaltet wird.
 4:    '    Achtung: Wenn eine Warnung erscheint, Änderungen
 5:    '    zunächst nur auf Sicherheitskopien durchführen!!
 6:    '
 7:    '    Geschrieben 1.8.1987 von
 8:    '
 9:    '    Frank Wübbeling
10:    '    Stadtlohnweg 33 /E 30
11:    '    4400 Münster
12:    '
13:    '    Last Update 23.10.87
14:    '
15:    Starter:
16:    Alert 1,"Bitte zu bearbeitende|Disk einlegen.",1,"Ok|Exit",A
17:    If A=2
18:      End
19:    Endif
20:    A$=Space$(512)
21:    ' Schaffe Raum für einen Diskettenblock
22:    V=Xbios(8,L:Varptr(A$),L:1,0,1,0,0,1)
23:    '    Lies den ersten Block der Diskette
24:    If V<0
25:      Alert 3,"Parameterblock nicht lesbar!",1,"Ok",A
26:      ' Fehler beim lesen
27:      Goto Starter
28:    Endif
29:    S=0
30:    For I=1 To 512 Step 2
31:      S=(S+Asc(Mid$(A$,I))*256+Asc(Mid$(A$,I+1))) And 65535
32:      ' Bilde die Quersumme über alle Daten des ersten Blocks
33:    Next I
34:    '     Wenn Summe=$1234, dann ist der Block ausführbar
35:    If S=&H1234
36:      Alert 3,"Bootblock existiert bereits!|Soll die Ausführung
          umgeleitet|werden?",1,"Ja|Nein|Exit",A
37:      If A=2
38:        A=0
39:      Endif
40:      If A=3
41:        Goto Starter
42:      Endif
43:      If A=1 And Asc(A$)<>96
44:        ' Seltener Fall: der Block ist ausführbar, aber der
            erste Befehl
45:        ' ist kein Branch - da kann man nichts machen
46:        ' Oder ist die Disk bereits auf 60 Hz?
47:        Alert 3,"Patchen kann nicht|durchgeführt werden!",1,
           "Exit",A
48:        Goto Starter
49:      Endif
50:      Sc=Asc(Mid$(A$,2))
51:      '   Sc enthält die Startadresse des alten Bootprogramms-2
52:    Else
53:      A=0
54:    Endif
55:    B$=Mid$(A$,3,6)
56:    B$=""
57:    Restore
58:    For I=1 To 6
59:      Read C
60:      B$=B$+Chr$(C)
61:    Next I
62:    ' Hier steht das Maschinenprogramm:
63:    Data &h42,&h38,&h82,&h0a
64:    '                   clr.b $ff820a.w
65:    Data &h4e,&h75
66:    '                   rts
67:    If A=1
68:      B$=Left$(B$,4)+Chr$(&H60)+Chr$(Sc-4)
69:      '                 wenn der Bootblock bereits ausführbar ist:
70:      '                 Branch zum Urprogramm ersetzt das RTS
71:    Endif
72:    A$=B$+Mid$(A$,7)
73:    ' Ersetzt die ersten 6 Bytes im Bootblock durch das
         Maschinenprogramm
74:    S=0
75:    For I=1 To 510 Step 2
76:      S=(S+Asc(Mid$(A$,I))*256+Asc(Mid$(A$,I+1))) And 65535
77:    Next I
78:    S=(&H1234-S) And 65535
79:    B$=Chr$(S/256)+Chr$(S And 255)
80:    A$=Left$(A$,510)+B$
81:    ' Berechnet ein Wort und fügt es am Schluß in den Bootblock
         ein.
82:    ' Damit die Quersumme $1234 ist (=> Block ausführbar)
83:    V=Xbios(9,L:Varptr(A$),L:1,0,1,0,0,1)
84:    ' Schreibt den ersten Block wieder zurück
85:    If V<0
86:      Alert 3,"Bootblock konnte nicht|geschrieben werden.",1,
          "Ok",A
87:      ' Fehler beim Zurückschreiben - vielleicht Diskette
           geschützt?
88:    Endif
89:    Goto Starter
```

res (Normalfall!), siehe nach, ob der erste Befehl ein BRA.S ist. Wenn ja, notiere die Anfangsadresse des alten Installationsprogramms. Wenn nein, so ist die Anfangsadresse nicht ermittelbar, brich ab.

3. Schreibe in die ersten 6 Bytes des Bootblocks das neue Installationsprogramm, nämlich:

clr.b $ff8201
rts

oder

clr.b $ff8201
bra.s alt

Je nachdem, ob ein altes Installationsprogramm schon existierte oder nicht, alt sei hier die relative Anfangsadresse des alten Programms.

4. Berechne das letzte Wort des Bootblocks so, daß die Summe aller Wörter $1234 ergibt.

5. Schreibe den Block wieder zurück auf die Diskette.

Bemerkungen:

Für die Diskettenoperationen beim Spielen mit dem Bootsektor sollten übrigens immer die XBIOS-Funktionen benutzt werden, da diese nicht die Disk-infos auf dem Bootsektor selbst benutzen.

Das Programm hält sich selbst nicht an die in Punkt (3) aufgestellte Forderung, daß ein Bootsektor immer mit BRA.S beginnen sollte, deshalb kann das Programm nicht zweimal auf dieselbe Diskette angewandt werden.

Frank Wübbeling

GEM

-Programme aus dem Auto-Ordner

Gelegentlich habe ich mich geärgert,daß es nicht möglich ist, ein Gem-Programm aus dem Auto-Ordner zu starten. Das AES ist zu diesem Zeitpunkt noch nicht initialisiert. Als fauler Zeitgenosse habe ich einen Trick gefunden, das Programm doch zu starten. Damit Sie verstehen, wie's geht, muß ich etwas weiter ausholen.

Nach Einschalten und Reset wird der Programmzähler des 68000er mit $FC0020 geladen und dann passiert grob (ich weiß, auch fein) folgendes:

1) Hardware initialisieren (Videoshifter,MFP usw.)

2) Speichertest (wieviel haben wir denn?)

3) Vektoren für Exceptions setzen (VBL, Traps, u.a. für die schönen Bömbchen)

4) DOS initialisieren, Datum setzen (immer 06.02.86)

5) Bootsektor Disk und Harddisk lesen und eventuell ausführen

6) Auto-Ordner-Programme ausführen (nur TOS-Anwendungen mit .PRG, sehr sinnig)

7) Desktop starten und Line-F-Emulator installieren

8) Eventuelle Accessories laden und starten

9) Desktop aufbauen

10) Auf Maus-Ereignisse warten (z.B.File angeklickt ?)

Wir sehen also, daß erst nach Nummer acht Gem-Programme gestartet werden können. Wie geht's denn nun ?

Im Auto-Ordner müßte ein Programm stehen, das bis kurz vor Nummer 10 lauert und dann sozusagen selber 'mausklickt'.

So macht's das Programm dann auch ungefähr. Der Schlüssel zu allen Tricks ist der Line-F-Emulator. Wie der funktioniert, kann man im Data-Becker Buch 'ATARI-ST-INTERN' nachlesen. So ziemlich alle AES-Aufrufe laufen über den Line-F-Vektor $2C.

Weiter in unserem Superhyperstartprogramm:
Die Abfrage der Tastatur am Anfang des Programms dient der Sicherheit von Harddisk-Besitzern (wie mir). Der ST-PD-Harddisk-Treiber, den ich benutze, bootet nämlich die Auto-Ordner-Programme von der Harddisk.

Wenn dann irgendwas nicht stimmt (Tippfehler,Denkfehler) und das Programm abstürzt, bootet sich die Harddisk zu Tode. Die Tastaturabfage prüft, ob die Alternate-Taste während des Startens gedrückt ist, und beendet das Programm sofort, wenn dem denn so ist. Diese Abfrage ist übrigens auch bei selbstprogrammierten Accessories Marke 'Bastlerglück' empfehlenswert.

* Achtung ! Beim Blitter-TOS geht die Funktion Kbdshift beim Start aus dem Autoordner nicht !!! Abhilfe: Zuerst Spacetaste drücken, danach sofort die Alternate-Taste.

Offensichtlich muß erst ein Keyboard-Interrupt auftreten, damit der Tastaturprozessor initialisiert wird. An der Speicherstelle $E61 steht normalerweise der Tastaturstatus, der in Register D0 von Kbdshift zurückgegeben wird. Beim Blitter-TOS steht dort erst was, wenn vorher eine Taste mit ASCII-Code gedrückt wird.

Da auf Systemvariablen zugegriffen wird, müssen wir jetzt die Routine 'Init' im Supervisormode ausführen. Ich habe dazu den Supexec-Befehl des Xbios benutzt.

'Init' schreibt die Konstante $0BFFFFFF in den Exceptionvektor der Line-F-Routine bei $2C. Eigentlich wollte ich –1($FFFFFF) schreiben, aber das Blitter-TOS nummeriert alle Vektoren frühzeitig durch, elfter Vektor ist also $0BFFFFFF. Dann wird eine Vertikal-Blank-Routine in den ersten Slot der VBL-Sprungleiste bei $4D2 eingehängt. Bei jedem Bildwechsel (Farbe 50/60 Hz, Monochrom 71 Hz), bei meinem Farbmonitor also bei 50 Hz alle 20 ms, wird 'neuvbl' ausgeführt. Diese Routine tut zunächst nicht viel mehr, als auf den Line-F-Emulator zu warten. Ist die Zahl $0BFFFFFF nicht mehr im Line-F-Vektor $2C, nehmen wir mal an, der Desktop ist gestartet und der Line-F-Emulator ist installiert.

Jetzt wird's haariger. Der VBL-Slot wird wieder gelöscht. Statt dessen hängen wir jetzt eine Routine 'neulf' vor den Line-F-Emulator. Diese neue Line-F-Routine lauert jetzt darauf, daß alle Accessories geladen und gestartet sind. Bei jedem Line-F-Sprung des AES wird der Opcode auf das Auftreten von $F08C getestet. Hinter $F08C versteckt sich der Multi-Event-Aufruf. Ist dieser Code aufgetreten, wird der Filename in die Dta $933A kopiert und die Startadresse $FE1DE2 nach A0 geladen. Dann wird einfach an diese Adresse gesprungen und die ganze Arbeit wie Laden, Relozieren, Starten dem AES bzw. dem übrigen Betriebssystem (gern) überlassen. Die Routine 'neulf' wird vorher natürlich wieder ausgehängt und die alte Originaladresse in $2C eingesetzt.

Hier also der Sourcecode der ganzen Geschichte, mühevoll dokumentiert: Ohne Haftung bei Bomben oder Schlimmerem !!!!!!!!!

Herbert Dampel

Listing

```
 1:  ********************************************************
 2:  *   Kuma Seka-Assembler *
 3:  ********************************************************
 4:  vblslot:           equ $4d2           ;erster vbl-slot
 5:  line_f:            equ $2C            ;line-f-exeptionvektor
 6:  b_evnt:            equ $f2c8          ;ab hier Blittertos-
                                            adressen
 7:  b_dta:             equ $bbae
 8:  b_starter:         equ $fe2698
 9:
10:
11:  start:
12:            move    #-1,-(a7)
13:            move    #11,-(a7)           ;kbshift
14:            trap    #13                 ;bios
15:            addq.l  #4,sp
16:            btst    #3,d0               ;bei alt-taste zum
                                            ende
17:            bne     dannich
18:
19:            pea     init                ;routine init im
                                            supervisor-mode
                                            ausführen
20:            move    #38,-(a7)           ;supexec
21:            trap    #14                 ;xbios
22:            addq.l  #6,sp
23:
24:            move    #0,-(sp)            ;kein fehler
25:            move.l  #[ende-start+$100],-(sp) ;anzahl zu reser-
                                            vierende bytes
26:            move    #$31,-(sp)          ;ptermres
27:            trap    #1                  ;gemdos
28:
29:  dannich:  move    #0,-(sp)            ;term
30:            trap    #1
31:
32:  init:     ;wird im supermode ausgeführt
33:
34:            tst.b   $fc0003             ;version testen:
                                            Blittertos?
35:            beq     normtos             ;nein
36:            lea     m_evnt(pc),a0       ;sonst adressen
                                            patchen
37:            move    #b_evnt,2(a0)
38:            lea     dta(pc),a0
39:            move.l  #b_dta,2(a0)
40:            lea     starter(pc),a0
41:            move.l  #b_starter,2(a0)
42:
43:  normtos:  move.l  #$0BFFFFFF,d0      ;blöde Zahl wegen
                                           Blittertos
44:            move.l  d0,line_f          ;nach $2c schreiben
45:            move.l  #neuvbl,vblslot    ;vbl Routine einhängen
46:            rts                        ;schon fertig,zurück nach
                                           ptermres
47:
48:  neuvbl:   ;wird bei jedem Vertikal-Blank ausgeführt
49:
50:            move.l  line_f,d0           ;inhalt $2c nach d0
51:            cmpi.l  #$0BFFFFFF,d0       ;noch drinn ?
52:            beq.s   nixda               ;wenn line-f noch nicht
                                            da nach nixda
53:            clr.l   vblslot             ;sonst vbl löschen,wird
                                           nicht mehr benötigt
54:            move.l  line_f,a0           ;addresse line-f nach a0
55:            move.l  a0,savelf           ;merken für rücksprung
                                           und ende nach start
56:            move.l  #neulf,line_f       ;eigene routine in line-f
                                           einhängen
57:  nixda:    rts                         ;vbl fertig
58:
59:  neulf:   ;wird vor jedem line-f-aufruf ausgeführt
60:
61:            move.l  2(a7),a0            ;line-f-befehlscode nach
                                            a0
62:  m_evnt:   cmpi.w  #$F08C,(a0)         ;sprung nach $fdd3f4
                                           (multi-event)?
63:            bne     altlf               ;nein,zur alten line-f
                                           zurück
64:            move.l  savelf,line_f       ;sonst alten line-f
                                           wieder zurück in $2c
65:  dta:      move.l  #$933a,a0           ;dta nach a0
66:            move.l  #startfile,a1       ;filenamebeginn nach a1
67:  loop:     move.b  (a1)+,(a0)+         ;nach $933a schreiben-
68:            bne     loop                ;bis 0 nach filename
69:
70:  starter:  move.l  #$fe1de2,a0         ;adresse für filestart
71:  jump:     jmp     (a0)                ;gem-programm starten
72:  altlf:    move.l  savelf,a0           ;alte line-f nach a0
73:            jmp     (a0)                ;hüpf
74:
75:  savelf:   dc.l 0                      ;alte line-f-routinen-
                                           adresse
76:  startfile:dc.b 'AUTOGEM\*.*',0        ;ordnername und file-
                                           ersatzname damit
77:                                        ;ein beliebig benanntes
                                           programm gestartet
78:                                        ;werden kann
79:  even
80:  ende:     equ *                       ;hier samma fertig
81:
82:  ********************************************************
```

Leserecke & Leserbriefe

Leser —> Leser

Tips zum Anwenderprogramm 'GEM':

1. Wenn man die rechte Maustaste drückt und gedrückt läßt, kann man nicht aktivierte Fenster wie aktivierte Fenster behandeln!

Red.: Normalerweise funktioniert dies tadellos, doch gibt es Probleme sobald man Programme aus einem Unterverzeichnis starten will. Dann greift das Desktop auf Laufwerk A: zu und findet in der Regel (es sei denn, Sie wollen das Programm von diesem aus Laufwerk starten) nicht den richtigen Pfad.

2. Wenn man eine Floppy-Station in ein Fenster zieht, wird die Diskette kopiert, die gerade in der Floppy-Station, welche man in das Fenster gezogen hat, drin ist.

Steffen Berendt, Meinersen

Leser —> ST

Probleme mit Signum und Fujitsu DL 3400

Wenn ich mit Textverarbeitungsprogrammen arbeite, dann hauptsächlich mit Signum!. Mit einem STAR NL-10-Drucker hatte ich damit keine Probleme. Nun habe ich mir einen Fujitsu DL 3400-Drucker gekauft, und damit fingen die Probleme an. Mit Signum gab es Fehlausdrucke, die laut Händler am Signum-Druckertreiber liegen. Können Sie mir nun einen Rat geben, wie der Treiber zu ändern ist?

Andreas Natusch, Lehrte

Red.: Nach Rückfrage bei Application Systems /// Heidelberg wurde uns versichert, daß dort für registrierte Kunden ein Druckertreiber für Signum! und den Fujitsu DL3400 existiert. Die Telefon-Nummer lautet: 06221/300002

Problem mit Funktionseingabe in einem GFA-Basic Programm

Mitte des Jahres bin ich von einem C64 auf einen Atari 1040F "umgestiegen". Als Mathematiklehrer interessiere ich mich vorwiegend für Anwendungsprogramme aus der gymnasialen Mathematik. Aus meiner "C64-Zeit" habe ich eine recht umfangreiche Sammlung selbst geschriebener Programme, die ich nun mit Hilfe des in Ihrem Verlag erschienen "GFA-Basic-Buch"s auf den ATARI-Rechner übertrage. In meiner Sammlung habe ich u.a. eine Anzahl recht komfortabler Programme zur Analysis: Funktionsplotter, Programme zur Kurvendiskussion ect. Einige dieser Programme habe ich schon auf den Atari übertragen. Nur ist es mir bisher noch nicht gelungen, einen Algorithmus zur Funktionseingabe im laufenden Programm zu erstellen, ich kann sie bisher nur im Listing vornehmen. Beim C-64 stand dazu in recht einfach zu handhabender Form der Tastaturpuffer zur Verfügung. Atari-Kenner (angebliche ?!) haben mir gesagt, daß es keinen Algorithmus dafür in GFA Basic gibt. Stimmt das?

Ulrich Temme, Selm

Red.: Eine Möglichkeit, Funktionen über den Tastaturpuffer des ATARI ST einzugeben, ist uns zur Zeit nicht bekannt. Wir möchten Sie aber auf einen Artikel, Formelberechnung mit GFA-BASIC, in der Februar-Ausgabe 1987 verweisen, in dem auf das Problem näher eingegangen wird. Dort wird sowohl die Methode mit CHAIN und MERGE, als auch eine rekursive Aufschlüsselung der Formel beschrieben. Diese Ausgabe können Sie, falls Sie sie nicht besitzen, beim Verlag nachbestellen.

Form–Keybd und Form–button

Endlich glaubte ich, das GEM etwas zu durchschauen, da entdeckte ich in den GEM-Libraries von Megamax V1.1 die AES-Funktionen

Nr. 55:
form–keybd (*char, int, int, int)
Nr. 56:
form–button (*char, int, int)

Zwar habe ich durch Disassemblieren von AESBIND.O herausgefunden, daß der *char Pointer jeweils ins adr–in Array, und die integerZahlen nacheinander ins int–in Array eingetragen werden, aber:

Was machen diese Funktionen? Sie sind nirgendwo dokumentiert. Sicher können Sie mir leicht helfen.

Christian Siebert, München

Red.: Wir können Ihnen tatsächlich weiterhelfen. Die Form–keybdRoutine ist quasi ein Filter, der alle Control-Codes wie Cursortasten oder die ESC-Taste übernimmt und ihre entsprechenden Funktionen durchführt.

Form–button dient z.B. zur Erleichterung bei der Programmierung einer eigenen Form-do-Routine und übernimmt die Arbeit, die beim Anklicken eines Objektes mit der Maus vollzogen werden muß.

Für weitere Information möchten wir Ihnen die ST-Ecke der Juli/August-Ausgabe empfehlen, da dort beide Routinen erklärt werden und mit Ihnen ein eigener Dialog-Handler programmiert wird.

Atari ST / Nutzung von 1st–Word Plus Dateien für den Fotosatz bzw. Lichtsatz

Auf meinem Atari ST benutze ich das Textverarbeitungssystem 1st–Word Plus. Ich möchte die hiermit erstellten Dateien für den Foto- oder Lichtsatz weiterverwenden können, um auf diese Weise das erneute Abtippen durch den Setzer zu sparen, was schließlich eine zusätzliche Fehlerquelle bedeutet. Wie müssen die Dateien vorbereitet werden (reine ASCII-Dateien ?), und wie können diese Dateien in die Satzgeräte eingespeist werden? Welche Fabrikate bieten sich an? Für eine möglichst detaillierte Anwort, was die Gerätekonfiguration – eventuell auch Software – angeht, wäre ich Ihnen dankbar. Vielleicht können Sie auch Setzereien (in NRW) nennen, die in der Lage sind, diese Dateien zu verarbeiten.

Ferdinand Ullrich

Red.: Prinzipiell ist zu diesem Thema zu sagen, daß nur reine ASCII-Dateien verarbeitet werden können. Dabei ist eventuell eine Konvertierung von Sonderzeichen notwendig, da ASCII nicht gleich ASCII ist. Z.B. muß für eine Anpassung an einen IBM-Rechner das 'ß' einen anderen ASCII-Code erhalten (beim ST mit 1st–Word ASCII(158) und beim IBM ASCII(225)). Ferner haben die meisten Setzer, die direkt ASCII-Dateien verarbeiten können, einen APPLE MacIntosh oder einen IBM-Rechner (XT oder AT), der dann die Daten an die Satzanlage weitergibt. Für die Zukunft wird bestimmt auch der ATARI ST mit seinen noch kommenden Desktop Publishing Programmen verwendbar sein. So ist zum Beispiel bei CALAMUS von DMC eine direkte Schnittstelle zu einer Linotype-Satzanlage geplant.

BIETE HARDWARE

ST-Uhr: nur 25 DM! Info: I. Lazaridis / Blumbergstr. 48 / 42-Oberh.1

1040, SM124, SC1224, GFA-Basic 1st WORD PLUS, 1st MAIL, DB-Master One, monoSTar PLUS, 1st Prop., Flight 2, Psion Chess, Barbarian, Starglider, u.v.m. Preis VB, Tel. 0711/815190

SF 354 -- Tel. 02251/3373

Akustikkoppler **Epson CX 21** incl. ST-Digi Talk, Kabel u. 2 DFÜ-Bücher 300,- Tel.: 0711/755711

Computer + Zubehör in Berlin, reelle Preise, anrufen 030/667277

MEGA ST 2, nagelneu! 2500 DM Tel.: 0681-814033 Patrick

SF314 günstig, originalverpackt wie neu DM 300,- 09231/61325

Atari-Floppy NEC 1036A-Netzteil Gehäuse-Datenkabel - Neu - 340,- Atari Mouse - Neu - 70,- 1stWord+ mit Nummer + Anleitung 120,- 02208/8537

SF354 DM 150 Tel.: 07225/75329

Drucker **SHINWA CP-80**, Centr. par. dt. Zeichen **DM 250,-** Tel: 089/7934920 bis 23 Uhr

Super-Verkauf!! neu abzugeben ★ alles mit 1 Jahr Garantie ★ 1040ST (m. Doppellaufwerk), Maus Monitor, gr. Software-Bibliothek ★★ äußerst preisgünstig ★★ VB, ab 19[00] Uhr: ☎ 09324/751

★★★ **Superangebot!!** ★★★ Eprommer für Atari ab DM 149 Floppylaufw. (720k/NEC) DM 385 **ABC-SYSTEMS ★ Schillerstr. 56 4432 Gronau ★ Info anfordern!**

Monitor (swBAS) ST Netzteil Maus RomTOS je 70,- DM 0209/777402

Farbmonitor Atari SC 1224!NEU! Keine 10 Betriebsstunden DM 650 originalverpackt! Tel: 08641-8687

Farbmoniotor **SC1224 DM 630,-** ☎ 07131/76804 ab 7.1.88 Mayer

260 ST, ROMTOS 1 MBYTE (!) VB 550,- G. Habedank 0241/806257

520 STM - SF354 - Maus - TOS im ROM - CSF-Gehäuse + Software, kompl. DM 850,- Tel. 0871/25971

ST-Floppy 2 St. NEC 1MB im **19"-Gehäuse** mit Netz! 0208/853274

Weide RAM 512KB, 150 DM S. Rogler, Ludwigsbrunn 103, 8673 Rehau

★TV Modulator★ 150,- 069-735523

Weide-Echtzeituhr f. 1040 STF, Neu, DM 80,-, Th. Pleitgen 5419 Arnshoefen

Verk. 1040ST + SC1224 + ST Pascal plus + Flight II (Org.) + 11 Disk (3M) m Softw. 2000,- ab 18h 0761/26303. Daniel Smeds, Goethestr. 6 7800 Freiburg.

BIETE SOFTWARE

■■ PUBLIC DOMAIN ■■ ■PD sortiert nach Anwendungs-■ ■bereichen u. ST-PD-Disketten■ ■Preise: ab DM 5,- ■ ■ Gratisinfo: **R & P** ■ Amselschlag 14 **8735 Ebenhausen**

ST-Public-Domain! 240 Disks, St. 5,-, ab 11 St. 4,-! Gratisliste! Floppy DS 349,-, Doppelfloppy 649,-, usw. O. Schäfer Hard- und Software, Soldiner Str. 4, 1000 Berlin 65 030/494 88 20

Original Programme wegen Systwechsel. Liste von 0221/638619

★★Public-Domain DM 5,00★★ ★PD-Angebot dieser Ausgabe★ ★ Einzeldiskette SS DM 5,00 ★ ★ Pakete 1-10, 11-20, usw. ★ ★ auf je 10 Disk. DM 40,00 ★ ★Porto u. Verpack. DM 3,00★ ★V.-Scheck o. NN (+DM 3,50)★ ★E. Twardoch, Kais.-Wilhelm-★ ★ Str. 88, 1000 Berlin 46 ★

Wordplus 2.02 Druckertreiber f. **STAR NL-10** siehe Anz. ST-Comp. 10/87. Mit 60 Kbyte Anleit. 30,- per Rech. K. Plüher, K.-F.-Friesenstr. 26, 4690 Herne 1. Demo?

★★★ **PD-Service** ★★★ Angebot wie PD dieser Ausgabe Einzeldiskette SS **DM 6,00** Doppeldiskette DS z. B. 1/2... 81/82 u.s.w. (aufeinanderf. und ungerade beginnend **DM 9,00** Pakete 1-10, 11-20, 21-30, 31-40 und so w. je **DM 38,00** auf je 5 Disketten DS Preis inkl. Disk zzgl. Porto u. Verp. DM 5,00 (Ausl. DM 10,00) NN plus 1,70 besser V-Scheck Lieferung erfolgt sofort, Neuheiten ab ca. 6. des Monats **I K S, Schönblickstr. 7 7516 Karlsbad-MU ab 18 Uhr 07202/6793**

■ PD-Disk nur 4 DM ■ ■ aus ST-Nr. 1 bis heute ■ ■ CHS Schneider 0281/61772 ■

FIBU, Buchführung für Freiberufler - Vereine - Privat, G+V, Bilanz, Kostenst., Abschluß, MWSt einf. Bedienung. Nur **69,- DM!!!** **FAKTURA,** Lager- u. Kundenverwaltung, Rechnung, Serienbrief, Umsatz. Nur **49,- DM!!!** **MITGLIEDER**-Verwaltung für Vereine u. Sonstige; Serienbrief, Beitrag, Mahnen, Abbuchen, Statistik etc. Nur **69,- DM!** Info: ★★ Ottmar Kuschek, Mülheimer ★★ Mühle, 5378 Blankenheim.

■ **Belegloser Zahlungsverkehr** ■ mit Terminüberwachung und Vormerkungen Tel. 07531/73781

Das Beste aus **Public-Domain**★ Gratis-Katalog ★ einseitig 5,50 ★ zweiseitig 6,50 DM ★ M. Schönfelder ★ Tlf.: 02954-1050

BS-Fibu (Neu) DM 790,-; Beckertext ST (Neu) V 1.3 DM 150,-; MWSt ausweißb. Tel. 06103/63201

Viele Original-Prg. wg. Systemwechsel abzugeben. Auch als Paket. Rolf Zens, 02181-490807

PD-Software ab 2,- DM Disk Angebot wie PD dieser Ausgabe Tel: 02721/2432

Ahnenverwaltung m. Listen u. Grafikausg. v. Vor- u. Nachfahren, 70 M Pfeiffer, Espen-5, 4047 Dormagen

Grundwortschatz Französisch 1700 Vok.+Lernprg(PD) nur 30,- (Scheck/bar, inkl. Disk + Versand) S. Becker, Frz.-Schubert-Str. 33 6095 Ginsheim 2

Protext, DeskAssist, K-Minstrel, NetRamDisk, IsGemDa, Beckertext 50%NP 09446/1022 od. 0941/53762

★STatistik★Datenanalyse f. ST★ Param. / nichtp.-Tests / Lineare + nichtl. Regression / Var.-Analyse Datenüb. aus bel. Datenbank / 149,- Info geg. DinA4 Rückumschl. Dr. Th. Rupprecht, 8520 Erlangen Halbmondstr. 3

Depot Deluxe (NP DM 498) zu **DM 250 / Boffin** Textverarbeitung (NP DM 398) zu **DM 200 / Starglider** DM 40 / M. Schmid, Schoenbuehlring 2, CH-6005 Luzern, Tel. (041) 44 37 79 ab 19 Uhr

Last-/Gutschriften per Diskette Datenbank beliebig ☎ 089/3518584

BRANDNEUE PD-SOFTWARE Fast täglich neue Programme, direkt vom Autor, die es nur bei mir gibt! Riesiges Angebot auf ca. 30 Seiten! **Alle Programme sind einzeln auswählbar!!!** Der absolute Hit: → NUR 1,5 PF PRO KBYTE ← Sie können eigene Disks senden, oder auf Qualitätsdisks bestellen: **SONY, FUJI, BASF je 1DD nur 3,- DM, 2DD nur 4,- DM!! VERBATIM ab 2,- DM** GRATISkatalog bei Arne Zingel, Vermehrenring 11a, 24 Lübeck 1

Wir haben Sie, die Software für Ihren ST. PD-Software für versch. Emulatoren auf ST. Erweitertes neues Angebot! Bitte 1,10 Rückporto beilegen. Carsten & Marcus Postfach 65 06 02, 2000 Hamburg 65

PUBLIC DOMAIN SOFTWARE Incl. Diskette 5,50 DM Atari Computer Club Reutlingen Tel. 07071/87482 18-20 Uhr

dBMAN + dBMAN + dBMAN ab 298,00 Anwendersoftware Info 17-21 h S&MC H.Keseling Fanny-Lewald-Ring5·2050Hmb80·040/7351931

★ **Atari ST Sensationen 1988!** ★ 400 PD-Disks zu Tiefstpreisen Riesiges Soft- & Hardwareangebot » **Weit über 200 PD-Spiele** « » **Gratis- & 99-Pfennig-Aktion** « PD-Software für Erwachsene u.v.m. Gratisinfo anfordern bei ■■■ **Ralf Markert** ■■■ ■■ **Balbachtalstraße 71** ■■ ■6970 LAUDA ☎ **09343/8269**■

CAD-Programme für Atari ST CAMPUS V1.2..... DM 998,- DRAFIX 1 .... DM 997,50 Josef Johler Computer Wiesentalstr. 39, 7996 Meckenbeuren, Tel. 07542/21848

■■■!! **PD jetzt DM 5,50** !!■■■ ■ PD dieser Ausgabe auf ■ ■ **Mark.-Disk SS je DM 5,50** ■ ■ P&V. DM 5,00, ab 20 St frei!■ ■■ **NEU!** ■ **NEU!** ■ **NEU!** ■■ ■ **Weg mit dem PD-Müll** - ■Jetzt eigene PD-Sammlung m.■ ■ **Bereichssortierung als Disk** ■ ■**o. einzel. PRG's nach KByte!**■ ■ **SUPER GRATIS-Katalog!!!** ■ ■J. Rangnow / 7519 Eppingen■ ■Talstr. 8/07262/5131 ab 17 Uhr■ ■■■ **PD-Express-Versand** ■■■

Prg-Aufruf über Pull-Down-Menü (s/w) DM 39,- Splines-Software Münchnerstr. 69, 8228 Freilassing

PD-Grafik-Bilder von Degas, NEC u. a. Katalog: Frey, Rheinstr. 12A, 6538 Münster-Sarmsheim

Lohnsteuerjahresausgleich/Einkommensteuer 87/88. Grunddaten + Ergebnis speichern. WasWäre-Wenn Lohn- u. ESt-Tabellen 87/88 30 DM Vork. J. Höfer, Grunewald 2a, 5272 Wipperfürth 02192/3368

Megamax C, Bücher „Einstieg in C" „Programmieren in C" DM 400 N. Schweighart, ☎ 089/988796

NEU! SPICE auf Atari ST. Volle GEM-Unterst. Grafik mit Zoom. Komf. Editor. Nur Fr. 99,- Excl bei KOINONIA LAB. Postf. 320 CH-9470 Buchs. Info gratis. **NEU**

P6 Zeichened. 25,- 089/3008622

ST-Vokabeltrainer 29,- Pull-down-Menüs + GEM-Technik Eingeben + Abfragen + Sortieren M. Schöttelndreier, Sendstraße 3 4600 Dortmund 12, T. 0231-200158

★★★★ **PD-Service** ★★★★ Gratisinfo anfordern bei: Roland Audritz ★ Braunschweiger Str. 21 ★ 3340 WF-Salzdahlum

PLOT_ST für Pro-FORTRAN! Einfachste Ausgabe grafischer Darstellungen auf Bildschirm und Drucker. Mehrere Funktionen gleichzeitig darstellbar. Kompatibel mit Großrechner Standard (Plot-79). DM 95,-. Handbuch DM 15,- (wird verrechnet); Info gratis. M. **Gamer,** Friedrichsring 26, 6050 Offenbach.

Verkaufe Aladin + Rom + PD-Softw. Telefon 06898/37979 ab 16[00]

Wärmebedarf DIN4701 + K-Zahl DM 110 ★ Rohrnetzber. DM 60 ★ Demodisk für beide Programme DM 10 Vorkasse von J. Binder Eichendorffstr. 15 · 5030 Hürth

ST-PD-Mailbox 0211-719261 8N1

„Das neue Vokabelprogramm:" Vocmaster, mehrfache Bedeutung möglich (Synonyms), GEM, Menülei. Grundwortschatz: 2000 Vokab. **49,-, Info: Gratis, Demo: 20,-** Reiner Kocher T: 089/3134946 Caracciolastr. 16, 8000 München 45

TKC-Video/WST-Englisch/Beckertexte! **Originale** für: 45/65/145 DM 0208/853274! **ST-Uhr: nur 25 DM!**

SUCHE HARDWARE

ATARI 520ST+, Floppy, Monitor s/w gesucht. 06622/1380 (abends)

Suche guterh. Monit. SM124 sowie SF314 o. Komp. biete Farbmon. Thomson RGB CM36382 Tel. 0711/535933 nach 18 Uhr

SUCHE SOFTWARE

MAXI, D4, M3 usw. 05321/26963

Suche in Verbindung mit 1040ST geeigneten Druckertreiber für Mannesmann-Tally MT140I! Bitte Info an: F. Wadsack 089/7232739

Suche Fletcher-Powell und maximum likelihood Programme zur Parameter-Anpassung mit MegaST Dr. Max Hahn, Humboldtstr. 10 4044 Kaarst 1

DTP-Prg. billig ges. T. 09306/1037

TAUSCH

Suche Kontakt zu Lehrern, die **Physikaufgaben in 1ST Word+** und **Physikprogramme in GFA-Basic** schreiben und sammeln. Tel.: 07082/20492

KONTAKTE

★ST-SOFTWAREENTWICKLUNG★ Schreibe Software, die speziell auf Ihre Bedürfnisse zugeschnitten ist. Kostenl. Info anford. Thomas Leufkes, Knappenstr. 10, 4353 Oer.-Erkenschwick Tel.: 02368/55020

VERSCHIEDENES

Original: Datenbank „IS, GEM, DA" DM 90,- Profi Painter ST 60,- SCANNER (Druckeraufsatz) 120,- Suche: NEC P6/7 **Tel. 02266/7894**

ATARI ST/MSX-Literatur Kat. kostenlos: H. Weidinger Postf. 21 05 46, 8500 Nürnberg 21

Joyce -> Atari, CP/M -> Atari Info: Bernd Drost, Schulstr. 67 6382 Friedrichsdorf 06175/604

Su. Hardware, Software, Bücher usw. f. Atari ST (0921/81384)

Verleihe Eprommer! 0251/26380

★ **20 neue Fonts für Signum** ★ ★ Professional Quality ★ ★ für 9-, 24- und Laserdrucker ★ ★H. Manschus Tel. 0651/76814★ ★Karl-Marx-Str. 70b · 55 Trier★



Nun auch in Deutschland!

– Software die es in sich hat!

Schwarz auf Weiß kann jetzt jeder auf seinem ATARI ST unter MS-DOS arbeiten. Doch nicht nur monochrom, nein, auch in Farbe ist nun der Zugriff auf die Welt der PC-Rechner möglich.

Die Software-Emulation **PC ditto** öffnet allen ATARI ST Anwendern das Tor zum gewohnten professionellen Business Standard.

Mit dem **PC ditto** können Sie mühelos mit Lotus 1 – 2 – 3 oder Symphony Ihre Kalkulationen erstellen.

Mit dem **PC ditto** können Sie Ihre Daten mit DBase III plus verwalten.

Mit dem **PC ditto** haben Sie Zugang zu dem schnellen Turbo Pascal Compiler.

Mit dem **PC ditto** läuft Ihr GW BASIC Interpreter, aber auch Borlands neuestes Kind Turbo Basic.

Mit dem **PC ditto** läuft auch die Software, die es für den ATARI ST noch gar nicht gibt.

Mit dem **PC ditto** laufen so viele Programme, daß wir sie hier gar nicht alle auflisten können.

Der **PC ditto** unterstützt die ATARI-Festplatte, den Druckerport, sogar den Laserdrucker und alle Schnittstellen, soweit es die Hardware des ST zuläßt.

Machen Sie aus Ihrem ATARI ST den preiswertesten PC-Clone! Bestellen Sie den **PC ditto** zum Super-Preis von nur DM 198,–.

Wir laden Sie ein! Steigen Sie mit Ihrem ATARI ST in die Welt des MS-DOS ein.



Hiermit bestelle ich

_ PC ditto für DM 198,–

Anruf genügt!
Tel.: 0 61 96 / 48 18 11, Mo–Fr 9–13 und 14–17 Uhr.
Schriftliche Bestellung nur gegen Vorauskasse oder Nachnahme (Versandkosten DM 7,50; bei Nachnahme zuzüglich DM 3,50 Nachnahmegebühr)

Name: ______________ Vorname: ______________

Straße: ______________ Ort: ______________

Unterschrift: ______________

Nähere Informationen gegen ausreichend frankierten Rückumschlag nur bei:



Industriestraße 26
6236 Eschborn

MS-DOS und GW-BASIC sind Warenzeichen von Microsoft Corp. Lotus 1 – 2 – 3 + Symphonie sind Warenzeichen von Lotus Dev. Corp. Turbo Pascal und Turbo Basic sind Warenzeichen von Borland Corp. DBase III Plus ist ein Warenzeichen von Ashton-Tate Corp.

Jahresinhaltsverzeichnis

ST Computer

Aktuelles

Softwaretest

Allgemeines

Business (Datenbanken, Tabellenkalkulationen, Handel, Fibu)

Grafik

Programmiersprachen

Textverarbeitung

Softwarelistings / Programmierpraxis

Hardware

Hardwaretest

Relax (Spiele)

Grundlagen

Kurse & Serien

ST-Ecke

Messen

Bücher

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

Anzeigenschluß für Heft 3/88: 22.01.1987

1000 Berlin



Computare
Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d

DATAPLAY
Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61





DIGITAL COMPUTER
Verkaufsbüro (1.OG) 1 Berlin 12
Knesebeckstr. 76 · Tel. 8827791
Software Hardware Beratung Zubehör Service Literatur



Keithstraße 26 · 1000 Berlin 30
26 111 26

1000 Berlin





Computershop Edith Behrendt
Fürbringerstraße 26 · 1000 Berlin 61
Telefon 0 30/691 76 66

ATARI
... wir machen Spitzentechnologie preiswert.
Vertragshändler
UNION ZEISS
Kurfürstendamm 57 • 1000 Berlin 15
Telefon 32 30 61

2000 Hamburg

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/494400

Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon Sa. Nr. 040/641 50 91

2000 Hamburg

Gerhard u. Bernd Waller GbR
Computer & Zubehör-Shop
Kieler Straße 623
2000 Hamburg 54
040/570 60 07 + 570 52 75

Hardware
Software
Beratung
Service



ATARI Systemfachhändler
Münsterstraße 9 · 2000 Hamburg 54
Telefon 040/56 60 1-1

NEU: Software Shop
RADIX Bürotechnik
Heinrich Barth Str. 13
2000 Hamburg 13
Telefon: 0 40-44 16 95



2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt
Heiligengeiststr. 20, 2120 Lüneburg
Tel. 04131/46122, Btx 402422
Mo.-Fr. 9[00]-18[00] und Sa. 9[00]-13[00]

2210 Itzehoe

Der Computerladen
Inhaber Ulrich Bübel Martin Kopplow
Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90/91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · ☎ 04 31 / 56 70 42

2350 Neumünster



Klosterstraße 2 · 2350 Neumünster
Telefon (0 43 21) 4 39 33

2390 Flensburg

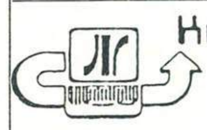


2800 Bremen

PS-DATA
Doventorsteinweg 41
2800 Bremen
Telefon 04 21 - 17 05 77

2850 Bremerhaven

HEIM- UND PERSONALCOMPUTER



Kurt Neumann
Georgstraße 71
2850 Bremerhaven
Tel. 04 71 / 30 21 29

HARDWARE · SOFTWARE · PAPIERWARE

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

2950 Leer



- HARDWARE-SOFTWARE
- SYSTEM-ENTWICKLUNG
- ORGANISATION
- EDV-SCHULUNG
- EDV-BERATUNG
- SERVICE-WARTUNG

Augustenstraße 3 · 2950 Leer
Telefon 04 91 - 45 89

3000 Hannover

COM DATA
Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36



Großer Hillen 6 · 3000 Hannover 71
Telefon (0511) 52 25 79

DATALOGIC COMPUTERSYSTEME
ATARI ST · COMPUTER · HARDWARE · SOFTWARE
BERATUNG · SERVICE · VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL. 0511 · 32 64 89

trendDATA Computer
IBM · EPSON · TRIUMPH ADLER
HEWLETT PACKARD · ATARI etc.
trendDATA Computer GmbH
Am Marstall 18–22 · 3000 Hannover 1
Telefon (05 11) 1 66 05-0

3040 Soltau

F & T Computervertrieb
Am Hornberg 1
(Industriegeb. Almhöhe)
3040 Soltau
Tel. 0 51 91 / 1 65 22

3100 Celle

Ludwig Haupt jr.
Büro-Einkaufs-Zentrum
Gerhard-Kamm-Straße 2
Ruf 8 30 45, Postfach 140
3100 Celle

3150 Peine

Wieckenberg & Schrage GmbH
Computertechnik
Hard- u. Software
Woltorfer Str. 8, 3150 Peine
Tel. 0 51 71 / 60 52/3 o. 0 51 73 / 79 09

3170 Gifhorn

COMPUTER-HAUS
GIFHORN
Braunschweigerstr. 50
3170 Gifhorn
Telefon 0 53 71 - 5 44 98

3300 Braunschweig

COMPUTER STUDIO
BRAUNSCHWEIG
Rebenring 49-50
3300 Braunschweig
Tel. (05 31) 33 32 77 / 78

3400 Göttingen



3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

3470 Höxter



3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler
Rudolf-Schwander-Str. 5-13
3500 Kassel
Tel. (05 61) 70 00 00

RUDOLPH
Computersysteme
& Bürotechnik
Einzel+Großhandel Im+Export
Telefon: 0561/472737
Telex Fax Btx **0561472739**
Computerstudio: 3500 Kassel
Frankfurterstraße 311
Versand: 3501 Hoof Pf. 1160

3550 Marburg

L W M
COMPUTER SERVICE
Bahnhofstraße 26 b
3550 Marburg/Lahn
✆ 0 64 21 - 6 22 36

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik – Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

HOCO
EDV ANLAGEN GMBH
Flügelstr. 47
4000 Düsseldorf
Tel. 02 11 - 77 62 70

4050 Mönchengladbach



Hindenburgstr. 249
4050 Mönchengladbach
Tel. 0 21 61 - 187 64

4150 Krefeld



- Service-Center
- Hardware
- Software
- Erweiterungen

COP
Computer Service GmbH
Lewerentz Str. 111
4150 Krefeld
Tel. 0 21 51/77 30 42

4200 Oberhausen

LaSch
das Buch und Software Haus
Inh. Rainer Langner u. Franz Schnitzler GbR

Nohlstraße 76 · 4200 Oberhausen 1
Telefon 02 08/80 90 14

4300 Essen

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (02 01) 17 63 99

4320 Hattingen



4330 Mülheim

BNS
Computer und Bürotechnik
Vertriebsgesellschaft mbH
Dickswall 79 4330 Mülheim Telefon 02 08/3 40 34

Computer Hard- und Software auch im Leasing
Computerkurse für Anfänger und Fortgeschrittene

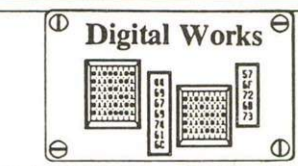
Tandon · SEL-Fernkopierer · VICTOR · NEC · ATARI · OKI

4400 Münster



COMPUTER SYSTEME GMBH
Daimlerweg 39 - 4400 Münster
Telefon 02 51 / 71 99 75 - 9

4410 Warendorf



Computer-Fachhandel – Hardware & Software

Jörg Kellert – Helmut Müller GbR
Brünebrede 17 · 4410 Warendorf
Tel. 0 25 81/6 11 26

4422 Ahaus

ATARI · Epson · Fujitsu
Molecular · NCR · Tandon · Schneider · Star
OCB

OCB-Computershop	OCB-Hard- und Software
Wallstraße 3	Wessumerstraße 49
4422 Ahaus	4422 Ahaus
Tel. 0 25 61/50 21	Tel. 0 25 61/50 21

4430 Steinfurt

ATARI SCHNEIDER STAR NEC SEIKOSHA PANASONIC EPSON

Computer
Büromaschinen
Service

Telefon 02551/2555

Tecklenburger Str. 27 · 4430 Steinfurt

4500 Osnabrück

Heinicke-Electronic
Kommenderiestr. 120 · 4500 Osnabrück
Telefon 05 41 - 8 27 99
Wir liefern Micro-Computer seit 1978

4600 Dortmund

Bürostudio BOLZ
Brauhausstraße 4 · 4600 Dortmund
Telefon 02 31 - 52 77 13 - 16



Atari, Genie, Schneider, Tandy, Brother, Star, Memorex, BASF, Verbatim
cc Computer Studio GmbH
Software-Hardware-Beratung
Service-Eilversand

Ihre Ansprechpartner:
v. Schablinski
Jan P. Schneider T. 0231/528184 · Tx 822631 cccsd
Elisabethstraße 5
4600 Dortmund 1



Elektronik
Computer
Fachliteratur
ATARI-System-Fachhändler
4600 Dortmund 1, Güntherstraße 75, Tel. (02 31) 57 22 84



4620 Castrop-Rauxel

R. Schuster Electronic
OBERE MÜNSTERSTR. 33-35 · ✆ (02305) 37700 · 4620 CASTROP-RAUXEL



System-Fachhändler

4650 Gelsenkirchen-Horst



Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststr. 15 · 4650 Gelsenkirchen-Horst
Tel. 0209/52572

4700 Hamm



4800 Bielefeld

hardware
software
organisation
service
CSF

CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106–108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

Carl-Severing-Str. 190
4800 Bielefeld 14

Telefon: 05 21 / 45 99-150
Telex : 9 37 340 krab d
Telefax: 05 21 / 45 99-123

MICROTEC
Software
Hardware
Beratung
Service

5000 Köln

BÜRO MASCHINEN
braun
AM RUDOLFPLATZ GmbH
5000 KÖLN 1
RICHARD-WAGNER-STR. 39
RUF: 02 21 / 21 91 71

5010 Bergheim



EDV-Beratung · Organisation
Programmierung · Home/Personal-Computer
Software · Zubehör · Fachliteratur
Zeppelinstr. 7 · 5010 Bergheim
Telefon 0 22 71 - 6 20 96

5060 Bergisch-Gladbach

Computer Center
Buchholzstraße 1
5060 Bergisch-Gladbach
Telefon 0 22 02 - 3 50 53

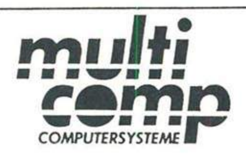
5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5200 Siegburg

Computer Center
Luisenstraße 26
5200 Siegburg
Telefon 0 22 41/6 68 54

5220 Waldbröl



Waldstraße 1 · 5220 Waldbröl
☎ (0 22 91) 44 08/33 86

5300 Bonn

Coco GmbH

Schumannstr. 2
5300 Bonn
Tel. 02 28 - 22 24 08

5400 Koblenz

SCHMITT
COMPUTERSYSTEME
Casinostraße 40
5400 Koblenz
✆ 02 61 - 3 65 28

5457 Straßenhaus

DR. AUMANN GMBH
Computer-Systeme
Schulstr. 12
5457 Straßenhaus
Telefon 0 26 34 - 40 81/2

5500 Trier



Güterstr. 82 · 5500 Trier
☎ 06 51 - 20 97 - 10
Fordern Sie unsere Zubehör-Liste an.

5540 Prüm

ATC COMPUTER
J. M. ZABELL
Ritzstraße 13 · Pf. 10 51
5540 PRÜM
– Tel.: 0 65 51 - 30 39 –

5600 Wuppertal

Jung am Wall

Wall 31—33
5600 Wuppertal 1
Telefon 02 02/45 03 30

5600 Wuppertal

MEGABYTE
Computer Vertriebs GmbH

Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT
Scheiderstr. 12 · 5630 Remscheid
Telefon (0 21 91) 2 10 33 - 34

5800 Hagen

ATARI
... wir machen Spitzentechnologie preiswert.

Vertragshändler **Axel Böckem**
Computer + Textsysteme

Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Tel. 0 23 31 / 7 34 90

5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

6000 Frankfurt

Müller & Nemecek
Kaiserstraße 44
6000 Frankfurt/M.
Tel. 0 69 - 23 25 44

WAIZENEGGER
Büroeinrichtungen

Kaiserstraße 41
6000 Frankfurt/M.
☎ 0 69 / 2 73 06 - 0

ATARI
... wir machen Spitzentechnologie preiswert.

Jetzt bei uns. Vertragshändler
bco
Büro-Computer + Organisations GmbH
Oederweg 7—9
6000 Frankfurt/M. 1
☎ (0 69) 55 04 56-57
Hardware * Software * Beratung * Service

SCHMITT
COMPUTERSYSTEME
Hanauer Landstr. 417
6000 Frankfurt/M. 1
Tel: 0 69 / 41 02 71
FAX: 0 69 / 42 57 71

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
✆ 0 61 51 / 5 60 57

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Elisabethenstr. 15 · 6100 Darmstadt
Luisencenter · Tel. 0 61 51 - 10 94 20

SCHMITT COMPUTERSYSTEME
Mühlstraße 76
6100 Darmstadt
✆ 0 61 51 - 2 45 74

6200 Wiesbaden

SCHMITT COMPUTERSYSTEME
Rheinstraße 41
6200 Wiesbaden
✆ 0 61 21 - 30 73 30

6240 Königstein

KFC COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6300 Gießen



Schneider ATARI Commodore
BAUMS
BÜRO · ORGANISATION
Bahnhofstr. 26 · 6300 Gießen
Telefon: 06 41 / 7 10 96

6300 Gießen



Thomas Heß
Asterweg 10 · 6300 Gießen
Telefon 06 41 / 3 91 53

6330 Wetzlar



Fachmarkt für Computer u. Unterhaltungs-electronic in Wetzlar,
Einkaufszentrum Bahnhofstraße, Tel. (0 64 41) 4 85 66

6400 Fulda

Schneider ATARI Commodore
WEINRICH
BÜRO · ORGANISATION
Ronsbachstraße 32 · 6400 Fulda
Telefon: 06 61 / 4 92 - 0

6457 Maintal

Landolt-Computer
Beratung · Service · Verkauf · Leasing
Wingertstr. 112
6457 Maintal/Dörnigheim
Telefon 0 61 81 - 4 52 93

6500 Mainz

:ELPHOTEC
Computer Systeme
Ihr Atari Systemhändler
mit eigenem Service-Center
Walpodenstraße 10
6500 Mainz
Telefon 0 61 31 - 23 19 47

SCHMITT COMPUTERSYSTEME
Karmeliterplatz 4
6500 Mainz
✆ 0 61 31 - 23 42 23

6520 Worms

ORION
Computersysteme GmbH
Friedrichstraße 22
6520 WORMS
Tel. 0 62 41 / 67 57 - 67 58

6700 Ludwigshafen

MKV Computermarkt
Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

MKV Computermarkt
Gilgenstraße 4
6720 Speyer
Telefon 0 62 32 - 7 72 16

6750 Kaiserslautern

C.O.S
Computer Organisation GmbH
6750 Kaiserslautern
Telefon (06 31) 9 20 51

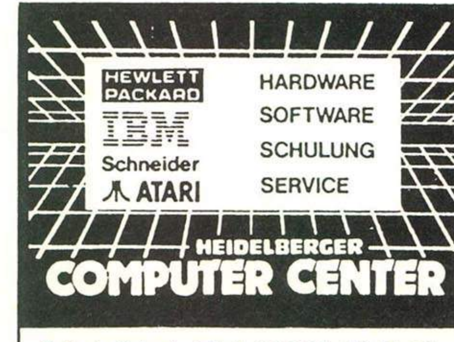
6800 Mannheim

GAUCH + STURM
Computersysteme + Textsysteme
6800 Mannheim 24
Casterfeldstraße 74-76
☎ (06 21) 85 00 40 · Teletex 6 211 912

Computer-Center
am Hauptbahnhof GmbH
L 14, 16-17
6800 Mannheim 1
Tel. (06 21) 2 09 83 / 84

6900 Heidelberg

JACOM COMPUTERWELT
Hardware · Software
Schulung · Service
Mönchhofstraße 3 · 6900 Heidelberg
Telefon 0 62 21 / 41 05 14 - 550



Bahnhofstr. 1 / Tel. 0 62 21 / 2 71 32

7000 Stuttgart

SCHMITT COMPUTERSYSTEME

Tübingerstr. 18
7000 Stuttgart



7022 L-Echterdingen

Autorisierter ATARI-System-Fachhändler

ATARI ST

Matrai computer

Matrai Computer GmbH
Bernhauser Str 8
7022 L.-Echterdingen
☎ (0711) 797049

7030 Böblingen

Verkauf – Service – Software



Norbert Hlawinka
Sindelfinger Allee 1
7030 Böblingen
Tel. 07031/226015

COMPUTER CENTER

7100 Heilbronn

Unser Wissen ist Ihr Vorteil

Walliser & Co.
Mönchseestraße 99
7100 Heilbronn
Telefon 07131/60048

Computer-Welt



Am Wollhaus 6
7100 Heilbronn
Tel. 07131-68401-02

7101 Flein

der **COMPUTERLADEN** von



in der Falterstraße
7101 Flein

Beratung, Service, Verkauf,
Software-Entwicklung
direkt beim Peripherie-Hersteller

7150 Backnang

Computer-Fans finden bei uns alles von:

commodore
Schneider
ATARI

Servicestation
Vertragshändler
Computer-Systeme
Software-Hardware

weeske
Das Elektrohaus am Nordring
Potsdamer Ring 10
7150 Backnang
Tel. 07191 1528

7410 Reutlingen

Computer-Shop
Werner Brock
autorisierter Fachhändler f. ATARI, Schneider, Oki

Federnseestr. 17 · 7410 Reutlingen
Telefon: 07121/34287

7450 Hechingen



Gesellschaft für Datenverarbeitung mbh

Computer · Drucker
Zubehör · Fachliteratur

Schloßplatz 3 · 7450 Hechingen
Telefon 07471/14507

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis

HEIM + PC-COMPUTERMARKT

HARDWARE · SOFTWARE · LITERATUR

SCHEURER

ATARI COMMODORE CUMANA DATA-BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON

7475 Meßstetten 1 · Hauptstraße 10 · 07431/61280

7480 Sigmaringen

Wir stellen uns für Sie auf den Kopf!

Wir führen:
Hardware - Software - Zubehör
Zeitschriften - Bücher

Wir sind autorisierte Fachhändler von:

ATARI · Schneider · Tandon

SOFT&EASY COMPUTER GMBH

7480 Sigmaringen
Am Rappgässle
Tel. 07571/12483

7968 Saulgau
Pfarrstraße 13
Tel. 07581/2598

7500 Karlsruhe

papierhaus erhardt

Am Ludwigsplatz · 7500 Karlsruhe
Tel. 0721-1608-16

7500 Karlsruhe

MKV GMBH

Kriegstr. 77
7500 Karlsruhe
Telefon 0721-84613

7530 Pforzheim

DM Computer GmbH
Hard- & Software
Durlacherstr. 39
Tel. 07231-13939
7530 Pforzheim
Telex 783248

7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk

In der Jeuch 3
7600 Offenburg
Telefon 0781/57974

7640 Kehl/Rhein



eigener Service · eigene Software

Badstrasse 12
Tel. 07807/822
Telex: 752913
7607 NEURIED 2

Filiale:
Hauptstrasse 44
Tel. 07851/1822
7640 KEHL/RHEIN

ELEKTRO-MÜNTZER GmbH

7700 Singen

U. MEIER
Computersysteme

7700 Singen-Htwl.
Am Posthalterswäldle 8
Telefon 07731-44211

7730 VS-Schwenningen

BUS BRAUCH & SAUTER COMPUTER TECHNIK

Villinger Straße 85
7730 VS-Schwenningen
Telefon 07720/380 71-72

7750 Konstanz

ATARI ★ PC's ★ SCHNEIDER

computer - fachgeschäft



Rheingutstr. 1 · ☎ 07531-21832

7800 Freiburg

CDS EDV-Service GmbH
Windausstraße 2
7800 Freiburg
Tel. 07 61 - 8 10 47



KARTÄUSERSTRASSE 59
D-7800 FREIBURG/BRST.
TELEFON 07 61 - 38 20 38

7890 Waldshut-Tiengen

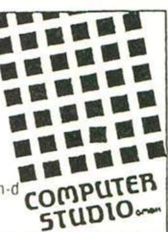
hettler-data service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 0 77 51 / 30 94

7900 Ulm

HARD AND SOFT COMPUTER GMBH
Ulms großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Herrenkellergasse 16 · 7900 Ulm/Donau
Telefon 07 31 / 6 26 99

EDV-Systeme
Software-erstellung
Schulung

Systemhaus:
Frauenstr. 28
7900 Ulm/Donau
Tel. 07 31 / 2 80 76
Telex 712 973 csulm-d



COMPUTER STUDIO

7918 Illertissen

biotech gmbh
technische Informationsysteme
Computerladen
Marktplatz 13
7918 Illertissen
0 73 03 / 50 45

7950 Biberach

HARD AND SOFT COMPUTER GMBH
Biberachs großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Schulstraße 6 · Bei der VHF
7950 Biberach · Tel. 0 73 51 / 1 22 21

7980 Ravensburg

GRAHLE
Expert Grahle Computer
Eisenbahnstr. 33
7980 Ravensburg
Tel.: 0751 / 15955

Vertragshändler für ATARI, Schneider und Star

8000 München

SCHMITT COMPUTERSYSTEME
Arnulfstraße
8000 München

8000 München



COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STR. 62 L
EURO-INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089 / 311 30 66 · TELETEX 898341

schulz computer
Schillerstraße 22
8000 München 2
Telefon (0 89) 59 73 39
Beratung · Verkauf · Kundendienst

8032 Gräfeling



COMPUTER SYSTEME SCHULUNG
Am Haag 5
8032 Gräfelfing
Tel. 089 - 8 54 54 64, 85 10 43

8100 Garmisch-Partenk.

Uwe Langheinrich Elektronik Center
Hindenburgerstr. 45
8100 Garmisch-Partenkirchen
Tel. 0 88 21 - 7 15 55
Bitte Gratisliste anfordern

8120 Weilheim

Klement
COMPUTER FACHHÄNDLER
Elektro-, Radio- und Fernseh-Center
Beleuchtungskörper · Schallplatten
Meisterbetrieb
8120 Weilheim · Admiral-Hipper-Straße 1
Geschäft ✆ 45 00 · Kundendienst ✆ 44 00
Computer-Studio ✆ 6 21 67
Interfunk-Fachgeschäft

8150 Holzkirchen

ATARI
Besuchen Sie uns!
Fordern Sie unseren Soft-ware-Katalog (520ST) an!

MÜNZENLOHER GMBH
Tölzer Straße 5
D-8150 Holzkirchen
Telefon: (0 80 24) 18 14

8170 Bad Tölz

Uwe Langheinrich Elektronik Center
Wachterstr. 3
8170 Bad Tölz
Tel. 0 80 41 - 4 15 65
Bitte Gratisliste anfordern

8330 Eggenfelden

Hot Space

Computer-Centrum
R. Lanfermann

Schellenbruckstraße 6
8330 Eggenfelden
Telefon 0 87 21 / 65 73

Altöttinger Straße 2
8265 Neuötting
Telefon 0 86 71 / 7 16 10

Innstraße 4
8341 Simbach
Telefon 0 85 71 / 44 10

8400 Regensburg

C-SOFT GMBH
Programmentwicklung & Hardware
Holzfällerstraße 4
8400 Regensburg
Telefon 09 41 / 8 39 86

Zimmermann elektroland

8400 Regensburg	8390 Passau
Dr.-Gessler-Str. 8	Kohlbruck 2a
☎ 09 41 / 9 50 85	☎ 08 51 / 5 20 07

8423 Abensberg

COMPUTERVERSAND WITTICH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 453



8500 Nürnberg



8500 Nürnberg

SCHMITT COMPUTERSYSTEME

Innere Laufer Gasse 29
8500 Nürnberg
✆ 09 11 - 20 97 17



8520 Erlangen



Computerservice Decker

Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76

Zimmermann elektroland

8520 Erlangen
Nürnberger Straße 88
Tel. (0 91 31) 3 45 68

8500 Nürnberg
Hauptmarkt 17
Tel. (09 11) 2 07 98

8600 Bamberg



8700 Würzburg

HALLER GMBH
Fachgeschäft für Mikrocomputer
Büttnerstraße 29
8700 Würzburg
Tel. 09 31 / 1 67 05

SCHOLL BÜROTEAM

Hardware · Software
Service · Schulung

computer center

am Dominikanerplatz
Ruf (09 31) 5 04 88

8720 Schweinfurt

Uhlenhuth GmbH

Computer + Unterhaltungselektronik

Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8900 Augsburg

Adolf & Schmoll Computer

Schwalbenstr. 1 · 8900 Augsburg
Tel. (08 21) 52 85 33 oder 52 80 87

Wir sind außerdem autorisierte Service-Fachwerkstatt für:

Schneider
commodore
ATARI

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Bürgermeister-Fischer-Str. 6-10
8900 Augsburg · Tel. (08 21) 31 53 - 416

SCHMITT COMPUTERSYSTEME

Frauentorstr. 22
8900 Augsburg
✆ 08 21 - 15 42 68

8940 Memmingen

EDV-Organisation Hard- + Software
Manfred Schweizer KG

Ulmer Str. 2, Tel. 0 83 31 / 1 22 20

8940 Memmingen

Österreich

A-1020 Wien

W + H.
Computerhandel Ges m.b.H.

Förstergasse 6/3/2 · 1020 Wien
Tel. 02 22 - 35 09 68

A-1040 Wien

Ihr ST-Fachhändler in Wien

Computer-Studio

Wehsner Gesellschaft m.b.H.

1040 Wien - Paniglgasse 18-20
Telefon 02 22 - 65 78 08, 65 88 93

A-8010 Graz



Schweiz

CH-1700 Fribourg

Softy Hard's Computershop
FRIDAT SA

Die ATARI ST Spezialisten

Grand Rue 42
CH-1700 Fribourg
Tel. 0041(0) 37 22 26 28

CH-2503 Biel

UE
URWA Electronic

Ihr ATARI ST Spezialist in der Schweiz.
0 32 / 25 45 53

Lindenweg 24, 2503 Biel

CH-3084 Bern

Computer Corner

Ihr ATARI ST-Partner in Bern

Hard-Software — Beratung
Midi-Schulung — Service

Seftigenstr. 240 — 3084 Bern-Wabern
Zentrum Wabern — Tel. 031/54 51 00

CH-3415 Hasle Rüegsau



CH-4054 Basel

Wir sind Amiga- und ST-Freaks.

Unsere Öffnungszeiten:
Von Dienstag bis Freitag
9.30 – 12.30 und 14.00 – 18.30 und
am Samstag 9.30 – 16.00



COMPUTER CENTER

Basel Tel. 061/39 25 25 · Holeestrasse 8 · 4054 Basel
Aarau Tel. 064/22 63 33 · Kasernenstrasse 26 · 5000 Aarau

CH-4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 062/63 17 27 + 63 10 27

CH-5430 Wettingen



Senn Computer AG

Zentralstrasse 93 — Tel. 056 / 27 16 60
CH-5430 Wettingen — Telex 814 193 seco

Hier könnte
Ihre Anzeige
erscheinen.

Anruf genügt:
Heim-Verlag
06151-56057

CH-8006 Zürich

ADAG Computershop

Universitätsstr. 25 · 8006 Zürich
Tel. 01/47 35 54

ATARI & WANG & EPSON

Computer-Center P. Fisch

Stampfenbachplatz 4
8006 ZÜRICH
01/363 67 67

CH-8021 Zürich



Langstrasse 31
Postfach
CH-8021 Zürich

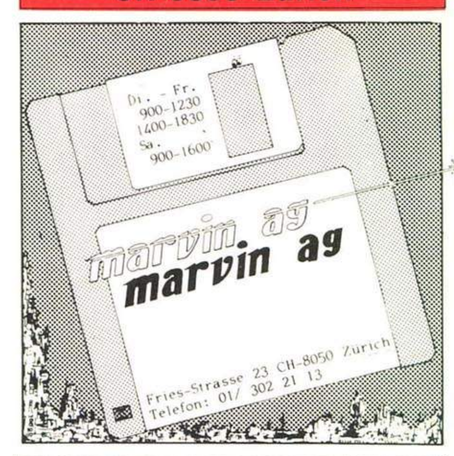
Tel. 01/241 73 73
Telex 814 193 seco

VILAN

Das Warenhaus der neuen Ideen

Ihr Computer-Fachhandel an der
Bahnhofstrasse 75 · Zürich

CH-8050 Zürich



CH-8052 Zürich



zentral in Seebach, Schaffhauser Str. 473, 8052 Zürich
Telefon: 01-302 26-00
Versandkatalog anfordern

CH-8200 Schaffhausen

ZIMELEC

CAR-AUDIO
COMPUTER + ELECTRONIC
Bachstraße 28 · 8200 Schaffhausen
Tel.: 053/5 52 24

Montag—Freitag	9.00—12.00
	13.30—18.30
Samstag	9.00—16.00

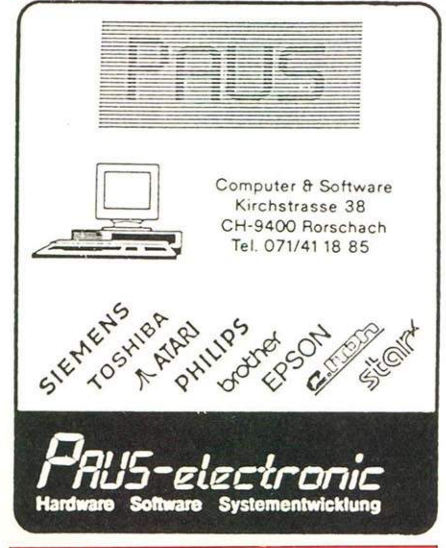
CH-9000 St. Gallen

VIDEO-COMPUTER-CENTER
GÄCHTER AG



Webergasse 22
9000 St. Gallen
Telefon 071/22 60 05

CH-9400 Rorschach



Luxemburg



Anzeigenschluß:
HEFT 2/88: 30.12.87
HEFT 3/88: 22.01.88



KLEINE HELFER

FESTPLATTEN-UTILITIES AUF DEM PRÜFSTAND

Die Festplatte als Massenspeicher wird immer beliebter. Grund: Die neuen Preisdimensionen, seit ATARI auch hier vorgeprescht ist. Das wissen natürlich auch die Softwarehersteller, und so quillt der Markt über vor allerlei *"unumgänglichen", "einmaligen"* und *"einzigartigen"* Utilities, die einem die tägliche Symbiose mit der Festplatte erleichtern sollen. Ob sie die ökologische Nische auf dem Schreibtisch wirklich heimeliger machen, mußten einige dieser Programme jetzt beweisen.

Vier Programme gingen an den Start, darunter zwei *Cache*-Programme, ein *De-Fragmentierer* und ein *Backup*-Programm. Für diejenigen, die im Festplattenmilieu noch neu sind, möchte ich jeweils vor der Programmbesprechung die zugrundeliegenden Ideen erklären.

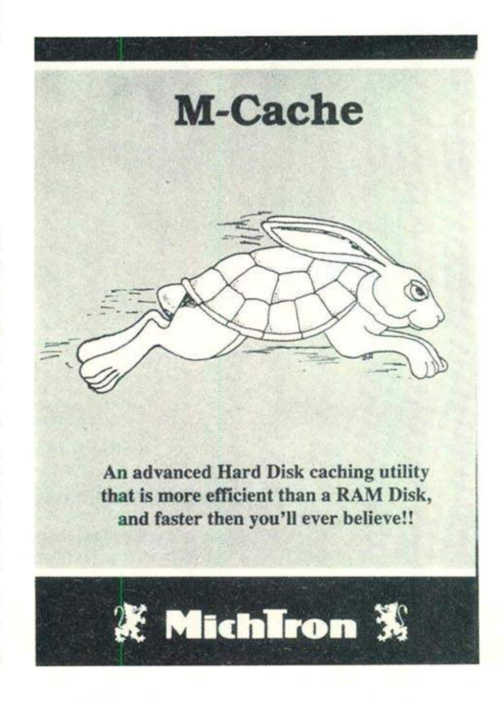
WAS IST EIN CACHE-SPEICHER?

Der Begriff *Cache* wird etwas inflationär gebraucht. Was die Festplatte betrifft, versteht man darunter ein Programm, das einmal von der Platte gelesene Sektoren eine Weile im Hauptspeicher puffert. Versucht man diesen Sektor erneut zu lesen, braucht man die Platte nicht mehr zu bemühen, sondern nur noch aus dem Speicher zu lesen, was natürlich viel schneller geht. Das bringt deswegen einen Geschwindigkeitsvorteil, weil die meisten Anwendungen immer wieder ganz bestimmte Bereiche auf der Platte ansprechen; vor allem die Verzeichnis- und FAT-Sektoren werden oft gebraucht. Zwar ist im GEMDOS ein Mechanismus eingebaut, der gerade diese Verwaltungssektoren puffern soll, doch funktioniert er wie vieles andere im GEMDOS nur höchst mangelhaft.

In dieser Kategorie gab es zwei Wettstreiter: den *Hard Disk Accelerator* von Beckemeyer Development Tools (deutscher Vertrieb durch ComputerWare) sowie *M-Cache* von Michtron. Beide Programme sind eigentlich schon etwas angegraut; der *Accelerator* stammt aus dem Jahre 1986, *M-Cache* ist sogar schon ein Veteran von 1985. Lassen Sie mich eine Münze werfen, mit welchem Programm wir anfangen... aha, *M-Cache* hat's erwischt.

KLEIN UND FEIN: M-CACHE

Timothy Purves, der Autor dieses Programms, dürfte vielen von seiner schon sehr früh erschienenen RAMDisk her noch ein Begriff sein. Er hat es sich einfach gemacht und puffert grundsätzlich nur *einzelne* Sektoren. Beim Aufruf der BIOS-Routine RWABS (und die wird bei den Cache-Programmen angezapft) kann man nämlich auch angeben, daß man meh-



rere Sektoren auf einen Schlag lesen will. Die relativ knifflige Verwaltung dieser Zugriffe hat der gute Timothy sich gespart; zwar meint er in seiner Anleitung, das sei Konsequenz aus TOS-Analysen, doch mag dabei auch ein Brocken Bequemlichkeit mitgespielt haben.

Wenn der Pufferspeicher voll wird, wirft *M-Cache 1 diejenigen Sektoren raus, die am längsten nicht mehr angesprochen wurden ("Least recently used"-Technik),* was einerseits recht einfach zu programmieren ist, wie ich freudig bei der Analyse des Programms feststellen durfte, andererseits wohl auch eine recht vernünftige Taktik ist. Beim Schreiben eines Sektors, der schon im Puffer ist, aktualisiert *M-Cache* den Puffer und schreibt ihn auch physikalisch auf die Platte, so daß keine Daten verlorengehen, wenn Sie wieder mal völlig genervt die RESET-Taste streicheln müssen oder bei Ihnen in der Nähe ein Kraftwerk in die Luft geht, und die Stromversorgung ausfällt (schon gut, ich weiß, bei Ihnen kommt der Strom aus der Steckdose).

Die Bedienung ist denkbar einfach: Auf der Diskette findet sich noch ein Programmchen namens MCCONFIG.TOS, dem man angeben kann, welche logischen Laufwerke gepuffert werden sollen; zum Schluß kann man noch bestimmen, wieviel Hauptspeicher für den Puffer geopfert werden soll (1 bis 999 Blöcke à *520* Bytes). Empfohlen wird ein Wert von 200 Blöcken, was etwa 100KB Speicher frißt (was würde wohl ein C64-Besitzer dazu sagen?).

Diese Voreinstellung speichert das Konfigurationsprogramm *im* eigentlichen Cache-Programm ab, der Code wird dabei also modifiziert, so daß man das Cache-Programm jetzt einfach in einen Auto-Ordner kopieren kann, wo es beim Booten seinen Dienst verrichtet, ohne noch extra irgendwelche Info-Files von der Diskette lesen zu müssen. Eine platzsparende Lösung, wie man überhaupt von *M-Cache* sagen muß, daß es sehr kompakt und geschickt programmiert ist (reines Assemblerprogramm); es wundert daher nicht, daß es auf der Diskette nur karge 1228 Bytes schluckt.

Man kann zwar die Installation von *M-Cache* durch Drücken von CONTROL, SHIFT oder ALT während des Bootens verhindern; garnicht gefallen hat mir dabei aber, daß das Programm dazu eine (wenn auch recht kleine) Warteschleife einlegt, die das Booten unnötig verzögert. Ich bin eben ein ungeduldiger Mensch.

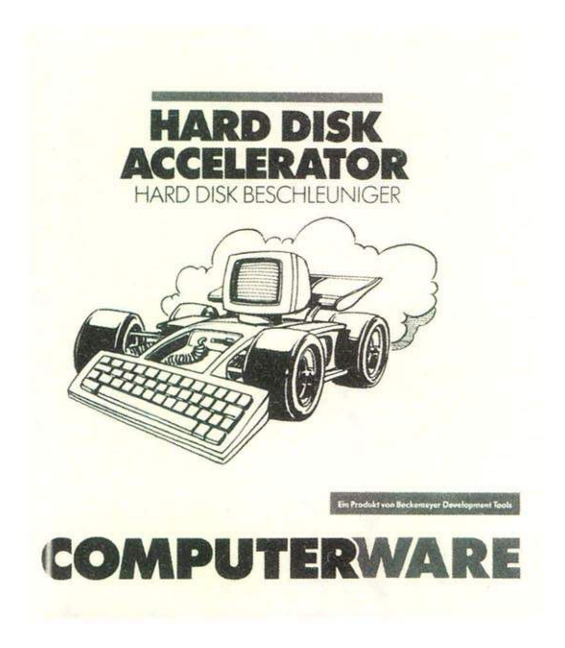
DER HERAUSFORDERER: HARD DISK ACCELERATOR

Bevor die Benchmarks klare Worte sprechen, lassen Sie mich noch das Konzept des Konkurrenten erläutern. Das Konfigurationsprogramm des *Hard Disk Accelerator* arbeitet GEM-unterstützt mit einer Dialogbox, in der auch angezeigt wird, welche Laufwerke überhaupt angeschlossen sind. Nur diese kann man auch zur Pufferung auswählen. Vorteil: Man kann nicht versehentlich ein Laufwerk puffern, das überhaupt nicht vorhanden ist. Nachteil: Man muß beim Installieren bereits die Konfiguration angeschlossen haben, die man auch für den Betrieb des *Accelerators* braucht. Die Puffergröße läßt sich nur in den Stufen 8, 16, 32 und 64 KB einstellen, was natürlich schon ein wenig einschränkt.

Leider war dem *Accelerator* keine Bedienungsanleitung beigefügt, obwohl sie auf der Verpackung angekündigt ist (sogar in deutsch, wohingegen die *M-Cache*-Anleitung in leicht verständlichem Englisch geschrieben ist). Wie *M-Cache* ist auch der *Accelerator* nicht kopiergeschützt, das ist bei einem Programm, das zur Installation auf einer Festplatte ja geradezu verdammt ist, nicht sonderlich lobenswert, sondern schlicht unerläßlich.

Aufgrund einiger Besonderheiten im Zeitverhalten vermute ich, daß auch der *Accelerator* nur einzelne Sektorzugriffe puffert. Für das Entfernen von gepufferten Sektoren aus einer übervollen Liste scheint er einen anderen Algorithmus als *M-Cache* zu verwenden.

Allerdings konnte ich das nicht mit Sicherheit herausfinden. Denn *Accelerator* ist offensichtlich in C geschrieben; der erzeugte Code ist jedenfalls C-typisch, nämlich etwas undurchsichtig, ekelhaft redundant und umständlich zu analysieren. Nicht zuletzt deswegen ist



das Programm auch recht aufgebläht (13 KB). Es geht auch ein wenig großzügiger mit dem Speicher um. Bei gleicher Puffergröße schnappt sich der 4 Accelerator etwa 29KB mehr vom Hauptspeicher als der kleine Konkurrent, was sich nur teilweise mit der Programmgröße erklären läßt. Auch zu jedem einzelnen Pufferplatz speichert der *Accelerator* etwas mehr Zusatzinformationen ab (14 Bytes gegenüber 8 Bytes bei *M-Cache);* diese Redundanz kann man aber getrost vernachlässigen.

DIE GLADIATOREN IN DER ARENA

Lassen wir sie also gegeneinander antreten. Als Prüfstein galt ein selbstgeschriebenes Programm (vgl. Test der HDplus-Festplatte von Vortex in diesem Heft), das in vier Phasen das zeitliche Verhalten der Platte testet. Ich will Sie nicht mit Abertausenden von Zahlen belästigen, darum das wichtigste Ergebnis im schlichten Klartext: *M-Cache* war schneller, und zwar bis zu dreißig Prozent. Vor allem beim Puffern vieler einzelner aufeinanderfolgender Sektoren zeigte sich diese Differenz. Die Anstrengung, in Assembler zu programmieren, lohnt sich eben immer.

Der *Hard Disk Accelerator* verlangsamte bei einem bestimmten Test sogar den Zugriff deutlich gegenüber einer ungepufferten (!) Festplatte: Ein Zeichen, daß die Verwaltung des Pufferspeichers nicht übermäßig gelungen ist.

Verzeichnisoperationen wurden mit beiden Cache-Programmen (in diesem Test) schon bei einer Puffergröße von 8 Kilobyte um etwa 20 Prozent schneller, andere Zugriffe brauchten zuweilen nur noch 15 bzw. 20 Prozent der Zeit *(M-Cache bzw. Hard Disk Accelerator)*, wenn der Puffer nur groß genug war.

Zum Spaß habe ich *M-Cache* auch einmal auf 999 Blöcke aufgeblasen, so daß ein ganzes Programmpaket komplett in den Puffer paßte. Nach dem ersten Einlesen ist dieser Puffer ähnlich fix wie eine RAMDisk, beim Schreiben sicherer. Allein, ich möchte meine RAMDisk nicht missen, schließlich kann man mit einem *Cache*-Puffer nicht kopieren, und nicht immer will man auf den Zeitvorteil beim Schreiben verzichten. Zudem puffern beide vorgestellten Cache-Programme nur Daten von der Festplatte (oder auch – ironischerweise – von einer RAMDisk). Ich würde einen kleinen Puffer von 20 oder 30 Sektoren empfehlen, um jederzeit die wichtigsten Verzeichnissektoren schnell greifbar zu haben.

Für *M-Cache* konnten wir leider keinen Preis erfahren, der *Hard Disk Accelerator* 119,– DM.

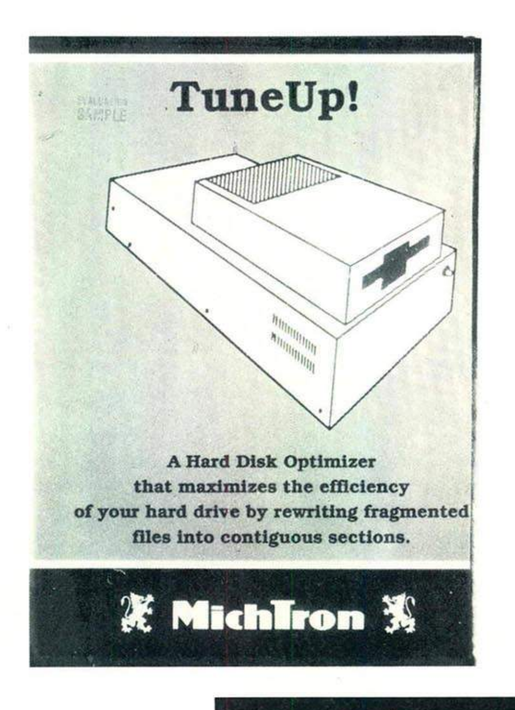
IN KLEINEN HÄPPCHEN SERVIERT

Normalerweise versucht TOS, für Dateien aufeinanderfolgende Sektoren zu belegen. Nun kann es aber – je voller eine Diskette oder Partition wird – dazu kommen, daß durch Löschen freigegebene kurze Bereiche für größere Dateien wiederverwendet werden, deren Datensektoren dann zwangsweise über die Diskette oder Partition zerstreut werden. Entnervende Folge: Längere Zugriffszeiten, weil der Lesekopf häufiger bewegt werden muß (und der ist mindestens so träge wie ich).

TuneUp, ebenfalls von Michtron, sucht diesen Mißstand zu beheben. Nach zwei verschiedenen Algorithmen werden Dateien wieder zusammengehängt; ebenso wird der freie Platz zusammengefügt, so daß die *Fragmentierung* (so nennt man das gerade beschriebene Phänomen) verschwindet.

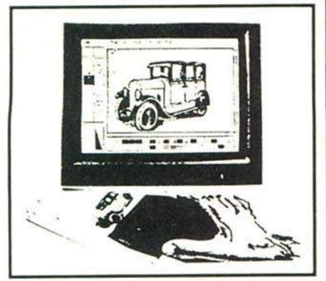
Man kann zwischen zwei Strategien entscheiden: Die eine versucht, Dateien unfragmentiert an den Anfang einer Partition zu verschieben. Die andere belegt das Ende einer Partition mit Dateien. Die erste Strategie ergibt optimale Lesezeiten, die andere garantiert, daß die Partition auch dann unfragmentiert bleibt, wenn Dateien gelöscht und neue erzeugt werden.

TuneUp ist auch nicht kopiergeschützt und läuft unter GEM denen man sich das Inhaltsverzeichnis des gesamten Dateibaums auf Bildschirm oder Drucker und statistische Daten über das Laufwerk geben lassen kann. Laut Anleitung prüft *TuneUp 1* auch auf "Konsistenzprobleme" in der FAT. Darunter werden offensichtlich Dateien verstanden, die zwar noch Einträge in der FAT besitzen, auf die aber kein Verzeichniseintrag mehr deutet. Werden solche Dateien gefunden, kann man sich im Wurzelverzeichnis einen Eintrag erzeugen lassen, der als Stiefvater für die verwaiste Datei fungiert.



Handy-Scanner

Begeistert von PC-Usern aufgenommen und eingesetzt, jetzt auch in höchster Vollendung für den Atari ST lieferbar !



bringt Graphik und Text superschnell auf Ihren Bildschirm !
- mit Software, Graphikpaket und deutschem Handbuch
- für Techniker, Leherer, Werbefachleute, Redakteure ...

Info anfordern! **898,--**

Public-Domain-Service

Wir liefern auf erstklassigem Diskettenmaterial die PD-Software **1-101** (siehe PD-Liste in dieser Ausgabe) und **100-142** (siehe unten ; nur bei uns erhältlich) ! **jede Diskette nur 8,--**

100. Diskettenverwaltungspr. 101. Digimusik Oxygen 102. TOS vom 2.6.86
103. Basicpr.,Terminalpr. 104. Pyramide, ST Pic 105. Finanzpr., Musikpr.
139. Ramdisks, Copy-Pr., Printhelper EPSON 140. Ramdisk, Spiele 141. Basic-Utilities, PrintDir
142. Proff

Aladin - Macintosh PD Software

auf über 30 Disketten für Atari ST in Verbindung mit Aladin in unseren Kopierdienst mit aufgenommen.
Nachfolgend ein kleiner Programmauszug:
Lisp • Terminalprogramm • Uhrenprogramm • Doc Doctor • Disk Librarien • Font Display • Ram Start • RCS Komplett •
Spiele: Tic Tac Toe • Solitaire • Go • Mathespiel • Wortspiel • Animationsprogramm • Mausefalle • "heiliges Programm •
Desk Acc.: Explorer • Mousepoint • Base to Base • Key Mouse
und viele mehr

Stellt nur einen Auszug aus Disketten dar!
!Aladin-Mac-Liste anfordern!

10er Blöcke: 10 Programme auf jeweils 5 Disketten 1DD (1MB)!!
1-10, 11-20, 21-20, 31-40, 41-50, 51-60, 61-70, 71-80, 81-90 und 101-110, 111-120, 121-130, 131-140 **je Block nur 40,--**

Monatsausgaben ST Jan/Febr, März/April **28,--**
Updates (U1, U2 ...) 8,-- **Liste anfordern !**

Atari-Schaltpläne

260 ST / 520 ST		29.80
520 ST+ / 520 STM		29.80
1040 STF		29.80
SF 314 / SF 354	je	19.80
SNM 804 / 1050	je	19.80
600 XL / 800 XL	je	19.80
SC 1224 / SM 124	je	19.80
Disk-**Royal** (Disk-Monitor)		89,-
Katpro-**Royal** (Diskettenverw.)		89,--
Wizard-**Royal** (Spiel)		139,--
Voc-**Royal** (Vokabeltrainer)		79,--
Ctask-**Royal** (Multitasking Bibl.)		89,--

Megamax C-Compiler

Ein komplettes Entwicklungssystem!
- Single Pass-Compiler
- Inline Assembler
- Disassembler
- Linker & Librarien
- Vollständige GEM-Libraries
- GEM-Editor, GEM-Shell
- 370 Seiten-Handbuch

Händleranfragen erwünscht **498,--**
deutsches Handbuch 49,--

Laufwerke für Atari ST (anschlußfertig)

- Komplett mit Metallgehäuse, Netzteil und Kabel
- Erstklassige Verarbeitung 100 % ST-Kompatibel !
- Mit TEAC od. NEC-Laufwerken

3,5"	Floppy	2 x 80 Spuren (1MB)	**348,--**
5,25"	Floppy	2 x 80 Spuren (1MB)	**448,--**
3,5"	Doppelfl.	4 x 80 Spuren (2 x 1MB)	**648,--**
3,5" + 5,25" Mix		4 x 80 Spuren (2 x 1MB)	**798,--**

Marconi-Trackball

Alternative zur Maus! Höchste Präzision mit geringem Platzbedarf. Neu! **nur 198,--**



Atari-Trackball nur 98,--

PAL-Interface II 298,--

Anschluß Ihres 260/520/1040 an jeden Farbfernseher. Erstklassige Bildqualität mit Ton aus dem Fernseher.
- Separates Netzteil
- Gleichzeitiger Anschluß von Monochrom und Colormonitor

Mouse-Pad 19,80

Gleitmatte für die Maus. Haargenaue Positionierung möglich

Programme und Zubehör

Tempus • Programmeditor	85,--
1st Word /Word Plus	99,--/199,--
1st Proportional	148,--
Wordstar / Mailmerge	199,--
Textomat ST	99,--
Protext ST	149,--
Becker Text ST	199,--
Write 90 • 90 Grad Drehung	79,--
Typesetter Elite • Seitenlayout	139,--
Megafont ST • Fonts + eigene	89,--
Art Direktor	173,--
Film Direktor	188,--
Monostar / Colorstar je	78,--
Maps & Legends	98,--
Degas	169,--
MICA	298,--
Animator	119,--
Modula II stand./ devel.	298,-- / 448,--
MCC Lattice C-Compiler	298,--
MCC Pascal Compiler	248,--
MCC Macro Assembler	158,--
MCC Make / Lisp	168,-- / 448,--
MCC BCPL-Compiler	329,--
K-Seka / K-Spread je	198,--
K-Graph / K-Comm	je 148,--
K-Resource / K-Word	je 118,--
K-RAM/Switch/Minstrel	je 89,--
Pro Fortran / Pro Pascal	je 448,--
DB Master one / Profimat	je 99,--
dBase II ST deutsch	348,--
Adimens ST neu	199,--
DB-Man deutsch neu	395,--
Trimm Base • Datenbank	298,--
Side Click	145,--
Copystar V 2.0 • Kopierpro.	169,--
Endlosetiketten 70x70 (200)	16,--
Disketten 1DD /2DD	39,-- / 49,--
Diskettenbox 40 /80	39,-- / 46,--
Druckerkabel	38,--
Abdeckung Konsole 520/1040	29,--
Abdeckung Monitor (124/1224)	39,--
Farbbänder, große Auswahl,	a. A.
CSF-Gehäuse für 260/520	148,--
ROM-Satz für alle ST´s	148,--
Akustikkoppler 300/1200	348,--
Twenty-Four Steinberg Musik	478,--
Atari SF 354	**198,--**
Star NL 10	**698,--**

Aladin (Mac Emulator)

Die Chance für Sie, auf Ihrem Atari ST Macintosh-Software zu fahren!
- höhere Bildschirmauflösung
- höhere Verarbeitungsgeschwindigk.
- integrierte, resetsichere Ramdisk
- Hard- und Software incl. Handbuch

398,--

Mac-Betriebssystem (ROM-Satz) **195,--**
Einzelinfo anfordern!

Kieckbusch Produkte

Logistix Kalkulation • Grafik • Datenbank (dtsch.)	398,--
Multi-Hardcopy Geniale Druckeranpassung	98,--
Desk Assist II+	98,--
STEVE Text-Grafik-Datenbank!	348,--
Graphic Artist CAD & Desktop Anwendung	698,--
Easy Draw	198,--
VIP Professional Kalkulation • Graphik • Datenbank (dtsch.)!	698,--

GFA-Produkte

GFA-Basic Interpreter Disk.	98,--
GFA-Basic Interpreter Modul	198,-
GFA-Basic Compiler Disk.	98,--
GFA-Objekt	198,-
GFA-Draft	198,-
GFA-Draft Plus	348,-
GFA-Sarter	58,--
GFA-Vektor	98,--
GFA-Publisher	398,-
GFA-Basic Buch	79,--
GFA-TOS & GEM Buch	49,--
GFA-Buch Progr.sammlung	29,--
GFA--Basic 68881	349,-

G-Data-Produkte

G-Ramdisk II	48,--
G-Diskmon II	89,--
Harddisk-Help & Extension	129,-
Interprint / Int. Ramd.	49,--/99,--
AS Sound Sampler	79,--

Omicron-Produkte

Omicron-Basic (Modul)	229,-
Omicron-Basic (Disk.)	179,-
Omicron Compiler (Disk.)	179,-
Omicron Assembler (ideal)	99,--

Campus	998,--
Demo-Version (wird angerechnet)	20,--
STAD	179,--
Publishing Partner • Desk-Top Publishing Programm	498,--
Fleet Street Editor • Desk-Top Publishing nur	348,--
Signum	448,--
Calamus	998,--
Word Perfect	910,--
Laser Brain	298,--

Weide-Echtzeituhr • Datum, Uhrzeit • Langzeit-Genauigkeit • Programm für Auto-Ordner DM 129,--

Speichererweiterung (512 K) DM 239,--

Monitor-Box • umschaltbar Monochrom/Color • mit 2 Cinch-Buchsen DM 58,--

Mouse Pad • Gleitmatte für Maus DM 19,80

Karl-Heinz • Potsdamer Ring 10 • 7150 Backnang
Telex 724410 weeba d • Kreissparkasse Backnang
(BLZ 60250020)74397 • Postgiro Stgt. 83326-707



Zahlung per Nachnahme oder Vorauskasse (Ausland per Scheck). Versandkostenpauschale (Inland 6,80 DM / Ausland 16,80 DM)

07191/1528-29 od. 60076

In den statistischen Angaben finden sich Hinweise, wieviele Dateien in wievielen Ordnern auf der Diskette zu finden sind, wie stark das Laufwerk belegt und fragmentiert ist, so daß man anhand dieser Angaben entscheiden kann, ob man den De-Fragmentierer auf die Diskette oder Partition loslassen will.

Einige Testläufe zeigten, daß *TuneUp* recht einfach zu bedienen ist (nicht zuletzt dank einiger im Programm integrierter Hilfsseiten) und auch funktioniert. Zumindest ging durch *TuneUp* nichts kaputt. Bombastische Geschwindigkeitssteigerungen waren nicht zu messen und sind auch nicht zu erwarten, weil die Fragmentierung meist einfach zu gering ist. Man kann sie auch in Grenzen halten, wenn man seine Festplatte nicht allzusehr füllt.

Wenn Sie mich fragen (sicher, mich fragt ja keiner, aber ich erzähle es trotzdem): Die De-Fragmentierung erledige ich lieber nebenbei, wenn ich von der Festplatte eine Sicherheitskopie anlege. Dazu braucht man nur ein Backup-Programm, das die Partitionen dateiweise auf Diskette sichert und nicht physikalisch (natürlich kann man auch selbst dateiweise per Desktop kopieren). Beim Restore-Prozeß wird die Platte dann so gefüllt, daß alle Dateien wieder zusammenhängend abgespeichert sind. Und Backups von der Platte zu ziehen, sollte man sich eh angewöhnen (wenn ich nur öfter auf mich hören würde...).

Mir gefällt *TuneUp,* für 79,50 DM käuflich zu erwerben, eher als Werkzeug, um die oben erwähnten verwaisten Dateien zu retten, und nützliche Statistiken über logische Laufwerke zu bekommen.

SICHER IST SICHER

Festplatten sind entsetzlich empfindlich; sagen zumindest deren Hersteller – und die Programmierer von Backup-Programmen. Denn je weniger man sich auf seine Platte verlassen kann und je gefährlichere Experimente man mit ihr macht, desto häufiger sollte man eine Sicherheitskopie auf Disketten machen. Natürlich kann man sich dazu auch einen *Streamer* kaufen (das ist ein schnelles Bandlaufwerk mit hoher Speicherkapazität), aber den meisten ST-Usern verbietet das wohl der Geldbeutel – schließlich muß man sich ja immer die neueste Ausgabe der ST-Computer leisten können.

Zunächst versucht wohl jeder, den Inhalt der Platte Datei für Datei per Desktop zu kopieren. Das geht für eine Weile recht gut; trotzdem verliert man dabei viel Zeit, wenn "Diskette voll" gemeldet wird, man eine neue Diskette formatiert, und dann erst feststellen muß, wo man denn vorher beim Kopieren aufgehört hatte, um die ganze Prozedur wieder von vorne zu starten. Manchmal möchte man auch nur bestimmte Dateien sichern, etwa nur diejenigen, die seit der letzten Backup-Aktion verändert wurden. Dabei leistet ein Backup-Programm gute Dienste und schont die Nerven (da kommt kein noch so verführerisch duftender Kaffee heran). Nebenbei kann man so auch das oben erwähnte Problem der Fragmentierung erschlagen.

Harddisk Utility von Volker Christen, im Vertrieb bei Application Systems Heidelberg, ist ein solches Backupprogramm und zeichnet sich durch einfache Bedienung und einige unübliche Features aus.

Zum Beispiel kann man sich ein Verzeichnis eines ganzen Dateibaums auf Drucker oder Schirm ausgeben lassen; blödsinnigerweise allerdings nicht von Laufwerk A oder B. Den Sinn dieser Beschränkung sehe ich wirklich nicht ein.

Auf der Diskette bekommt man auch ein kleines Prögrämmchen mitgeliefert, das sich um die 40-Ordner-Krankheit des ST kümmert. Vielleicht haben Sie schon davon gehört: Ab etwa 40 angelegten Ordnern auf der Festplatte häufen sich allerlei seltsame Erscheinungen; das liegt an einem zu kleinen internen Pufferspeicher, der durch das beiliegende Programm etwas vergrößert wird. Bei den MegaSTs (genauer, im BlitterTOS) ist dieser Fehler leidlich behoben, da können Sie sich diesen Handstand also sparen.

Harddisk Utility stellt praktisch ein eigenes Desktop zur Verfügung, auf dem man wie gewohnt Laufwerkicons anklikken, verschieben und öffnen kann. Um eine Partition zu sichern, öffnet man deren Verzeichnis, markiert dann die Dateien, die man auf Diskette bannen will, und zieht das Icon der Partition auf das Diskettensymbol von Laufwerk A oder B. Recht einfach und logisch also.

Einfachheit ist aber nicht der einzige Vorzug dieses Programms. Das Markieren von Dateien muß man nämlich durchaus nicht vollständig per Hand durchführen; in den Drop-Down-Menüs sind Optionen versteckt, mit denen man entweder alle Dateien (eventuell nach einer bestimmten Dateimaske, in der Joker erlaubt sind) markiert oder nur diejenigen, die seit der letzten Sicherung verändert wurden.

Jede Datei hat in GEMDOS ein sogenanntes Fileattribut, in dem der Dateityp kodiert ist. Ein Bit des Attributs ist eigens für Sicherungsaufgaben vorgesehen. Backup-Programme setzen es, um anzuzeigen, daß diese Datei bereits gesichert wurde. Beliebige Schreibzugriffe durch GEMDOS auf die Datei löschen das *Archiv-Bit,* wie man es auch nennt; daran kann ein Backup-Programm also erkennen, daß die Datei geändert wurde und damit vor dem Schlund des Hades gerettet werden muß.

Harddisk Utility (nebenbei gesagt, ein wahnsinnig einfalls-

reicher Name) fordert je nach Fortgang der Kopiererei neue Disketten an. Sind sie noch nicht formatiert, holt das Programm es nach (ebenso, wenn eine entsprechende Option in einem Drop-Down-Menü gesetzt wird). Dabei wird ein recht intelligentes selbstgestricktes Format verwendet; man kann bis zu 82 Spuren und 10 Sektoren auf zwei Seiten formatieren lassen, wobei noch ein Trick angewendet wird (elfter Sektorvorspann), um auch auf Systemen ohne eingebautes FASTLOAD diese Disketten besonders schnell lesen zu können. Natürlich geht aber in punkto Kapazität beim Backup nichts über HYPERFORMAT-Disketten.

Eine wichtige Besonderheit: Die Daten können auch komprimiert auf die Diskette gerettet werden. Das spart Platz, macht aber auch die Vorankündigung des Programms, wieviel Disketten man brauchen wird, etwas unsicher, weil das Programm ja vor dem Kopieren nicht genau weiß, wieviel Bytes es durch Komprimieren sparen kann. Überhaupt ist die Kompressionsmethode eine recht primitive (sie erkennt Bytewiederholungen und kürzt sie ab), so daß bei normalen Programmdateien nur wenig herauszuholen ist. Nur bei Grafiken und ähnlichen Dateien zeigt sich der Vorteil. Ansonsten kostet die Kompression mehr Zeit, als die paar eingesparten Bytes wert sind.

Das *Harddisk Utility* verzichtet auf die normalen Verwaltungsstrukturen auf den Sicherungsdisketten (deswegen kann man die Dateien darauf auch nicht vom Desktop aus ansehen); die Dateistruktur wird am Ende des Kopiervorgangs in einer eigenen Datei abgelegt.

Auch der umgekehrte Kopiervorgang (Restore) von der Diskette zurück auf die Platte ist sehr einfach zu bedienen. Überhaupt braucht man in diesem Fall die Anleitung kaum mehr als einmal zu lesen (was auch kein Genuß wäre, weil sie typisch deutsch, will sagen, ziemlich trocken geschrieben ist und keine großen Geheimnisse verrät).

Zum Preis von 69,– DM bekommt man ein solides Backup-Programm. Mißfallen hat mir allerdings, daß man Backup-Disketten des *Harddisk Utility* nicht per Desktop lesen kann; sicher, das bringt einen Zeitgewinn. Aber man stelle sich nur vor, man möchte den Inhalt der Platte auf Diskette retten und dort damit weiterarbeiten. Das *Harddisk Utility* kann man dazu nicht verwenden, weil man ja die Backup-Disketten nicht ordnungsgemäß lesen kann. Man kann natürlich versuchen, den Inhalt einer solchen Diskette per Restore in eine RAMDisk zu kopieren, was mit ein bißchen gutem Willen auch klappt, aber den Auf- und Umstand könnte man sich eben auch sparen.

(Claus Brod)

Hard Disk Accelerator:
Computerware Gerd Sender
Moselstr. 39
5000 Köln 50
Tel.: 0221/392583

TuneUp: Gerhard Knupe GmbH & Co KG
Güntherstr. 75
4600 Dortmund 1
Tel.: 0231/527531

Harddisk Utility: Application Systems /// Heidelberg
Englerstr. 3
6900 Heidelberg
Tel.: 06221/300002





ABO

ABO

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim-Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Einzelheft- u. Monatsdisketten Bestellung



Einzelheft- u.
Disketten Service

Absender
(Bitte deutlich schreiben)



Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim-Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Kleinanzeigen



Kleinanzeigen

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim-Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Abonnement

Ja, bitte senden Sie mir die ATARI-Computer Fachzeitschrift ab ______
für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 70,– frei Haus.
(Ausland: Nur gegen Scheck-Voreinsendung DM 90,– Normalpost. DM 120,– Luftpost)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Name

Vorname

Straße/Nr.

PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM______ liegt bei.

☐ Vorauskasse per Post-Einzahlung (Zahlkarte)

Garantie:
Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.

Datum Unterschrift

Datum Unterschrift



ABO



Einzelheft- u. Monatsdisketten Bestellung

ST-Computer können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 7,– (zuzüglich Gebühr für Porto und Verpackung) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli/Aug.	Sept.	Okt.	Nov.	Dez.

1986 = DM
1987 = DM

+ Gebühr für Porto u. Verp. = DM

☐ Scheck in Höhe zus. DM liegt bei

Disketten Service

Alle Programme, die in ST-Computer veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 ST-Computer-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten

Preis je Diskette 28,– DM	Jan./Febr.	März/Apr.	Mai/Juni	Juli/Aug.	Sept./Okt.	Nov./Dez.
	87	87	87	87	87	87

Lieferung: gegen beigefügten Scheck zuzügl. 5,— DM Versandkosten, unabhängig der bestellten Menge.



Einzelheft- u. Monatsdisketten Bestellung



Kleinanzeigen-Auftrag

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik

Biete an ☐ Hardware ☐ Software
Ich suche ☐ Hardware ☐ Software
☐ **Tausch** ☐ **Kontakte**
☐ Verschiedenes ☐

30 Buchstaben je Standardzeile – incl. Satzzeichen und Wortzwischenräume.
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen.

Bearbeitung **nur gegen Vorausscheck** über den entsprechenden Betrag (keine Überweisung)

☐ privat = DM 7,– je Zeile incl. MwSt.
☐ gewerblich = DM 15,– je Zeile + MwSt.
☐ Chiffregebühr = DM 10,–

☐ Scheck über DM______ ist beigefügt

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

Datum Unterschrift

Absenderangaben auf der Rückseite nicht vergessen

Kleinanzeigen

Kontaktkarte

Kontaktkarte

Bitte Adresse der Firma, bei der Sie Informationen, oder etwas bestellen möchten, auf der rechten Seite eintragen → → → → → → → → → →

Absender
(Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Postkarte

Bitte freimachen

Firma

Straße/Postfach

PLZ Ort

Kurzmitteilung

Kurzmitteilung

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Telefon

Postkarte

Bitte freimachen

Merlin Computer GmbH
ST-Computer Redaktion
Industriestraße 26

6236 Eschborn



PD Bestellung

PD Bestellung



Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Postkarte

Bitte freimachen

Merlin Computer GmbH
ST-Computer Redaktion
Industriestraße 26

6236 Eschborn

ST COMPUTER

☐ Ich bitte um weitere Informationen

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in ST-Computer Heft ______ Seite ___

Menge	Produkt/Bestellnummer	DM	gesamt DM

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)

Abgesandt am:

ST COMPUTER

Firma:

Bemerkungen:

ST COMPUTER

Meine Meinung

Zu dem Artikel ____________ in Heft ____, Seite ____
hätte ich folgendes zu bemerken:

☐ Ich möchte Ihnen folgendes Programm zur Veröffentlichung anbieten: (Kurzbeschreibung, Sprache, Länge in Druckerseiten, GEM/TOS)

☐ Ich kann über folgendes Thema berichten: (Tips & Tricks am ST, Hardware, Software, etc.)

☐ Ich möchte gerne Autor in der ST-Computer werden. Meine Fachgebiete: (z.B. LISP, Pearl, Modula-2, Assembler ...)

☐ Ich möchte, daß folgendes Public-Domain Programm in Ihre Sammlung aufgenommen wird.

☐ Sonstiges

Bei weiteren Angaben oder Fragen wenden Sie sich bitte schriftlich oder telefonisch an die Redaktion. Tel. 0 61 96/48 18 11

ST COMPUTER

Kurzmitteilung

ST COMPUTER

PUBLIC DOMAIN SERVICE

Ich bestelle folgende PD-Disketten:
(Siehe PD Service in dieser Ausgabe)

Zahlung erfolgt:

☐ per Scheck

☐ per Nachnahme

Je Diskette fügen Sie bitte einen Betrag von DM 10,– bei,
für Porto und Verpackung je Sendung DM 5,– (Ausland DM 10,–)

Datum

Unterschrift

ST COMPUTER

PD Bestellung

Datum

DELA Elektronik

Drucker

Star NL 10 inkl. Interface	DM **529,–**
Epson LX 800	DM **539,–**
Epson LX 800 f. C 64	DM **619,–**
Seikosha SL-90 AI	DM **798,–**
DELA-Printer Parallel	DM **509,–**
NEC P6	DM **1.049,–**

Monitore

NEC Multisync	DM **1.449,–**
Philips 8833	DM **678,–**

Disketten (100er Pack)

5"25 MD 1D	DM **75,–**
5"25 MD 2D	DM **85,–**
3"5 2DD	DM **250,–**

In unseren Filialen zahlen Sie entsprechende Preise auch für den 10er Pack!!!

Diskettenlaufwerke-Festplatten

Diskettenlaufwerk f. Atari ST	DM **399,–**
Amiga-Diskettenlaufwerk 3"5 Disk	DM **329,–**
Seagate ST225 inkl. Controller 20 MB	DM **598,–**
Seagate ST238 inkl. Controller 30 MB	DM **698,–**
Seagate ST251 40 MB 40 ms	DM **838.–**

Modems & Akustikkoppler & BTX

(Modems o. FTZ-Nummer)

IBM-Modem Hayes-kompatibel	DM **299,–**
Dataphon S 21-23d	DM **298,–**
BTX-Term f. IBM	DM **278,–**

Wir führen Commodore-Chips, Eproms, elektronische Bauteile u.v.m.
Fordern Sie unsere Preisliste an.
Außerdem führen wir reichhaltiges Zubehör für Commodore-, Atari-, Schneider- und IBM-Computer – Info unbedingt anfordern!

Sound Sampler für Amiga

Neben einer professionell gefertigten Hardware erhält das Komplettpaket eine komfortable Software.
Abspeichern der Daten als Objektfile.
Generierung von Sound-Disketten.
(Option: mit 2 Laufwerken kann unbegrenzt lange und ohne Unterbrechung gesampelt werden.)
Komprimiermöglichkeit (spart bis zu 50% Speicherplatz).
Erstellung von IFF-Files möglich; lädt jedes File (nicht nur IFF-Format).
Programm erkennt und unterstützt auch Speichererweiterungen.
Weitere Optionen: das Mixen der Sounds.

Das Alles naürlich zum DELA-Preis von DM **89,–**

Sound Sampler Software

Für alle, die bereits die entsprechende Hardware besitzen.

DELA-Preis DM **49,50**

DELA-AMIGA 500-EPROMMER

In Kürze bei DELA erhältlich: der Eprommer für den AMIGA 500.
Natürlich mit den gleichen überragenden Leistungsmerkmalen wie der DELA-ATARI-ST EPROMMER.

Natürlich zum DELA-Preis: DM **149,–**

DELA-Atari ST Epromdisk

Vereinigt die Geschwindigkeit einer Ramdisk mit der Zuverlässigkeit von ROM's.
Voll Software-gesteuert durch mitgeliefertes Treiberprogramm und Modulgenerator.
Auch für AUTO-Ordner und ACC-Dateien.
Die Grundversion von 512 K-Byte kann durch ein Aboveboard auf 1 M-Byte erhöht werden.
DIE GESCHWINDIGKEIT DER EPROMDISK IST HÖHER ALS DIE EINER FESTPLATTE!!!
Natürlich mit ausführlicher Anleitung für ein kinderleichtes Handling und zum

DELA-Preis DM **99,–**

DELA-Shugate-Interface für Amiga

Jetzt wird der Anschluß von 3"5 und 5"25 Laufwerken leichtgemacht.
Einfach mit Laufwerk und Computer verbinden – fertig!!!
Auch für Doppellaufwerke geeignet!

DELA-Preis DM **49,50**



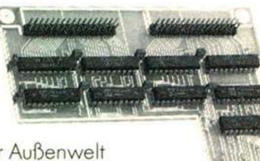
DELA-ATARI-ST EPROMMER

Technische Daten:

- Programmierspannung: 21 V und 12.5 V Softswitch
- kein Netzteil erforderlich
- vollkommen Software-gesteuert keine unprofessionellen Schalter
- echter hochwertiger TEXTOOL Sockel
- eingebauter Generator für Atari-Rom-Module
- Epromtypen: 2764, 27128, 27256, 27512, 27513, 27011 natürlich auch alle kompatiblen Typen und die entsprechenden CMOS, sowie A-, B-, C-Typen
- Normal-Mode, Schnell-Mode, Einzelbyte-Mode …
- wird einfach in den Romport gesteckt, der Druckerport bleibt natürlich frei

und die absolute Sensation:
der vorläufige Preis für dieses SUPERDING: DM **149,–**
(in Worten: eins-vier-neun DM!!!)

DELA-Atari ST-Userport



Die Schnittstelle zur Außenwelt (z.B. für Steuerungsaufgaben).
Leistungsmerkmale:

- 2 Steckleisten (insgesamt 64 bit I/O)
- je Steckerleiste 2x8 bit Input und 2x8 bit Output
- TTL-kompatibel und gebuffert
- In allen Programmiersprachen frei programmierbar!!!
- Ausführliche Anleitung mit vielen Beispielen

DELA-Preis DM **99,–**

Besuchen Sie uns mal in:

Köln 1 · Maastrichter Straße 23
Essen 1 · Schützenbahn 11–13 DGB-Haus Porscheplatz
München 22 · Bürkleinstraße 10

Fordern Sie unseren Katalog an – für alle Informationen – mit allen Preisen.

DELA immer aktuell, immer preiswert:

Am besten gleich bestellen!



24 Stunden Anrufbeantworter 02 21-7 15 17 30
Mailbox: 02 21-7 15 17 40
Telefon: 02 21-7 15 17 / 20-22
Telefax: 02 21-7 15 17 60
Teletex: 2 214 248 DELA

Versandbedingungen:
Versand ab 40,– DM, sonst nur gegen Vorkasse.
Nachnahmeversand NN-Spesen 7,50 DM, b. Vorkasse 4,00 DM.
Auslandsbestellungen nur gegen Vorkasse + 15,00 DM Versandkosten. Wir liefern auf Ihre Rechnung und Gefahr zu den Verkaufs- und Lieferbedingungen des Elektronikgewerbes. Die Preise gelten im Handel als unverbindliche Preisempfehlung.
Preis: Stand 15.12.1987

Bestellcoupon ausfüllen, auf Postkarte kleben oder im Umschlag an:
DELA Elektronik · Merkenicher Straße 87–89 5000 Köln 60

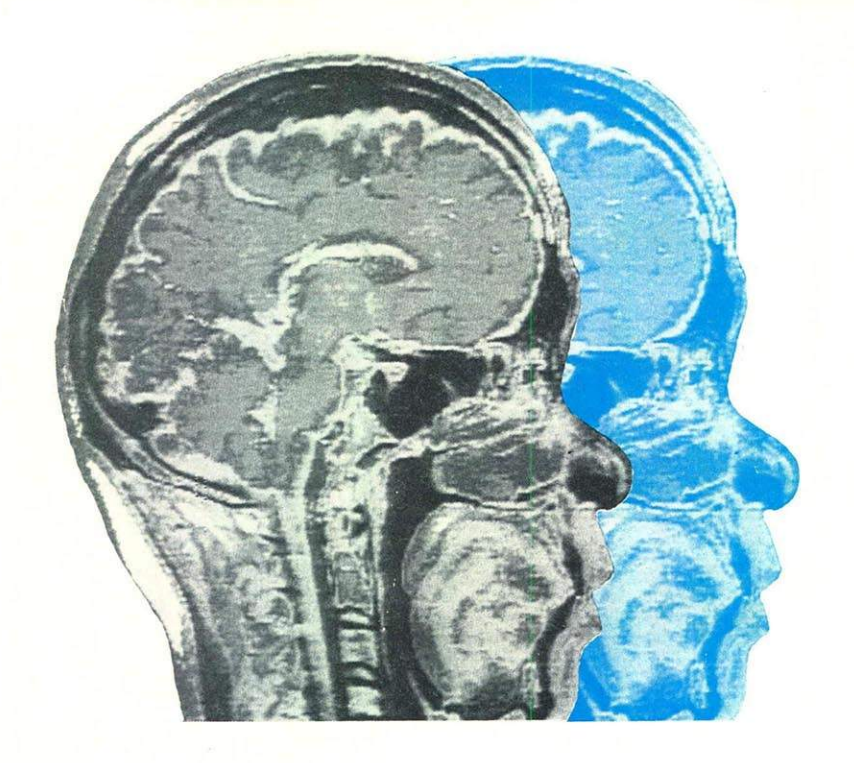
Bestellcoupon

St. Computer 1/88

Menge	Artikel	Einzelpreis

Name ______ Anschrift ______

Kunden-Nr. ______ Datum ______ Unterschrift ______



Elemente der künstlichen Intelligenz

Eine Einführung in Programmiermethoden und Sprachen der KI 8. und letzter Teil

Implementierung von Rahmen

In diesem letzten Teil der Serie über künstliche Intelligenz, möchte ich Sie mit dem Konzept des Rahmens vertraut machen. Wie ich bereits in einer der ersten Folgen dieser Serie betont habe, erkennt man Intelligenz auch daran, daß aus unvollständigen Informationen richtige Schlüsse gezogen werden, bzw. für die jeweilige Situation nützliche Aktionen eingeleitet werden. Gerade aber wenn die Wissensbasis unvollständig ist, bekommt ein Verfahren besondere Bedeutung, das als analoges Schließen bekannt ist.

Analoges Schließen bedeutet ganz einfach, daß das Programm die Kenntnis aus benachbarten Wissensgebieten benutzt um in der konkreten Situation Informationsmangel zu kompensieren. Besonders im Bereich des Lernens ist analoges Schließen von elementarer Bedeutung. Kein Wunder also, wenn die KI-Forscher nach Mechanismen gesucht haben, die analoges Schließen auch dem Computer ermöglichen sollen.

Rahmen dienen der Organisation von Wissen

Um dem Computer Analogieschlüsse zu ermöglichen, muß das zu vergleichende Wissen in einer geeigneten Form gespeichert werden. Hierzu sind die Rahmen (engl.: frames) besonders geeignet. Ein Rahmen ist nichts anderes als eine vereinheitlichte Property Liste. Und zwar besteht die Propertyliste aus dem Rahmen, Schlitz (engl.: slot) und Merkmalen (engl.: facettes). Hat man nun zwei Rahmen mit ähnlichen Eigenschaften, so können Informationen aus dem einen Rahmen in die entsprechenden Merkmale des anderen Rahmens vererbt werden. Wir wollen diese Eigenschaften von Rahmen am Beispiel des Programmes FRAME.LSP erarbeiten, welches ich in Anlehnung an [–] für XLISP geschrieben habe (Listing 1). Eine vollständige Implementation einer Übungssprache (PFL: Paedagogical Frame Language) ist bei [–] gegeben. Über die Einordnung von Rahmen in den KI-Bereich kann man bei [–] nachlesen. Kümmern wir uns also zunächst einmal um die Propertyliste, die alles beherrschende Datenstruktur unseres Programms.

Die Propertyliste

In Lisp gibt es eine Liste, in der man Eigenschaften eines Objektes ablegen kann, eben die Propertyliste. Diese ist nicht zu verwechseln mit Assoziationslisten, die man in beliebiger Anzahl erzeugen (und an Symbole binden) kann. Der Zugriff auf die Propertyliste des (X)Lisp Interpreters erfolgt über die Funktionen **SETF** und **GET**. Die Syntax dieser beiden Funktionen lautet:

(get Xsym Xeig) Holt den Wert der Eigenschaft **eig** des Symbols **sym** von der Propertyliste.

<**sym**> Das Symbol. <**eig**> Die Eigenschaft des Symbols, welche einen Wert hat.
(setf <zugr> <wert>)
Diese Funktion gibt den Wert **wert** an die in der Zugriffsform **zugr** enthaltene Eigenschaft **eig** des Symbols **sym** der Propertyliste.
<**zugr**>
Die Zugriffsform. Sie muß, wenn evaluiert, den Wert der

Eigenschaft des Symbols auf der Propertyliste erzeugen. In unserem Fall also (get **<sym>** **<eig>**).

<wert>

Der Wert, den die in der Zugriffsform enthaltene Eigenschaft **eig** des Symbols **sym** auf der Propertyliste erhalten soll.

An einem Beispiel wird der Sachverhalt leicht klar. Das Symbol **KLOTZ** mit der Eigenschaft **FARBE** soll den Wert **ROT** erhalten:

(setf (get 'klotz 'farbe) 'rot)

Dies ist die Zugriffsform. Sie reproduziert später wieder den Wert der Eigenschaft **FARBE** des Symbols **KLOTZ** aus der Propertyliste.

Anschließend kann mit der Zugriffsform der Wert wieder aus der Propertyliste geholt werden:

(get 'klotz 'farbe)

ROT1

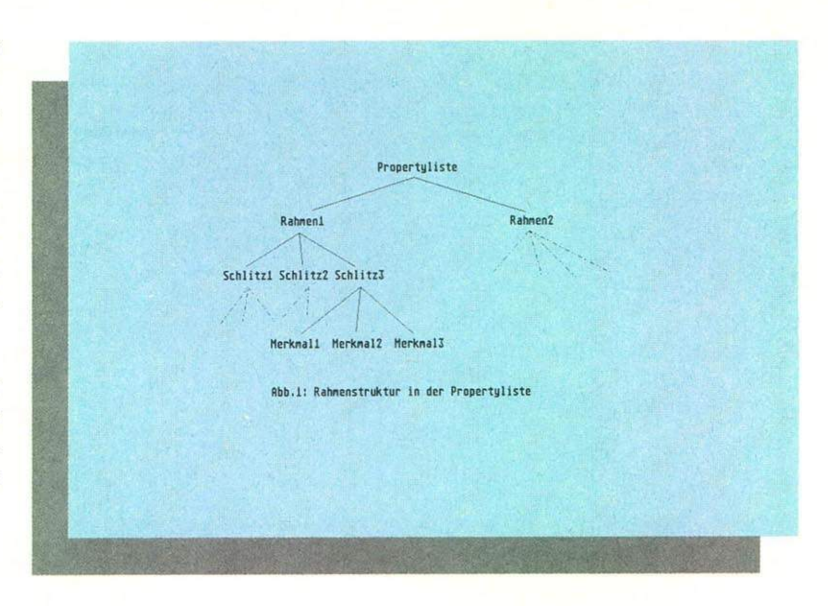


Abb.1: Rahmenstruktur in der Propertyliste

In XLISP gibt es noch zwei spezielle Zugriffsfunktionen für die Propertyliste: **REMPROP** und **PUTPROP**. Die erste entfernt eine Eigenschaft aus der Propertyliste eines Symbols, die zweite ist äquivalent zur Funktion **SETF**. Wir benutzen aus Kompatibilitätsgründen aber die Funktion **SETF**.

Wie wir bereits gelesen haben, ist ein Rahmen nichts anderes als eine normierte Propertyliste mit den Elementen **RAHMEN, SCHLITZ, MERKMAL**. Abb. 1 zeigt den Aufbau eines Rahmens in der Propertyliste. Natürlich ist die Anzahl der Rahmen in der Propertyliste, bzw. die Anzahl der Schlitze im Rahmen oder die Anzahl der Merkmale eines Schlitzes nur durch den verfügbaren Speicherplatz des Computers beschränkt.

Nehmen wir einmal an, wir wollten unser Wissen über Computer in Rahmen fassen. Dann könnte der Rahmen für den ATARI ST Computer vielleicht so aussehen, wie Abb. 2 es zeigt. Der Name des Rahmens ist **ATARI**, der erste Schlitz in diesem Rahmen ist die Typenbezeichnung **MODELL** mit dem Merkmal **WERT 1040ST**. Der zweite Schlitz des gleichen Rahmens ist **SPEICHER**. Standardmäßig wird er 1040ST mit 1Mb Speicher ausgeliefert. Der Besitzer obigen Gerätes hat aber sein Gerät auf 2Mb aufgerüstet. Dies ist erkennbar an den beiden Merkmalen **STANDARD 1MB** und **WERT 2MB**. Auch für den Schlitz **MONITOR** sind zwei Merkmale vorhanden **STANDARD SW** und **WERT COLOR,** woraus zu entnehmen ist, daß dieser Computer entgegen dem Standard mit einem Colormonitor ausgerüstet ist. Nun bleibt nur noch die Frage, wie man solch einen Rahmen einrichtet, bzw. die Informationen aus dem Rahmen abruft und weiterverarbeitet? Und wie funktioniert analoges Schließen mit den Rahmen?

Informationen holen

Es ist am leichtesten, aus der Propertyliste Informationen eines bestehenden Rahmens zu holen. Deshalb zunächst also die Zugriffsoperationen. Nehmen wir an, daß die Propertyliste den Rahmen aus Abb.2 enthält. Wir erhalten den kompletten Rahmen, wenn wir den Befehl **(get 'atari 'rahmen)** eingeben. Man erhält dann das Resultat aus Abb.3.

Wollen wir nicht den ganzen Rahmen, sondern nur ein Merkmal eines bestimmten Schlitzes eines bestimmten Rahmens, dann müssen wir uns eben durch die gegebene

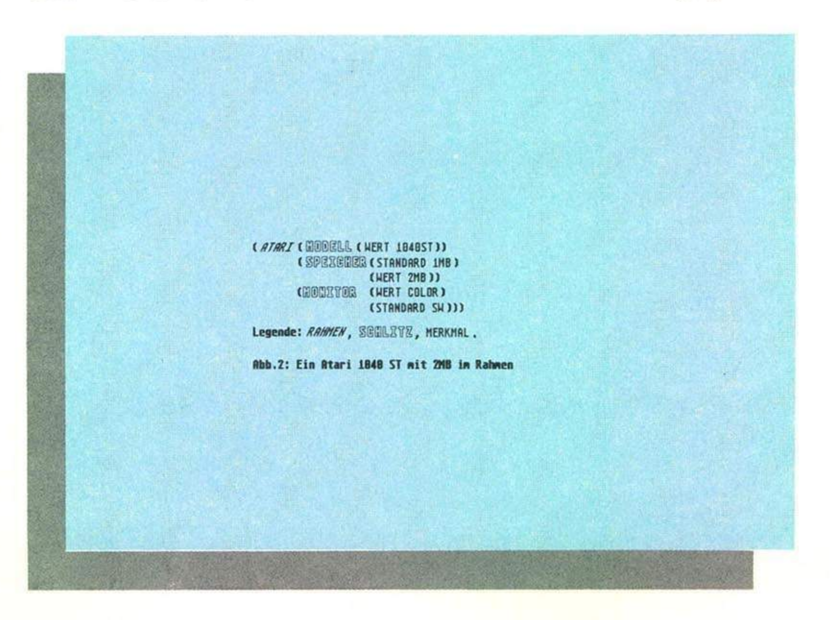


Abb.2: Ein Atari 1040 ST mit 2MB im Rahmen

Rahmenliste vorarbeiten. Inzwischen sind wir ja schon zu geübten XLisp(l)ern geworden, sodaß wir ohne Mühe in der obigen Liste eine sogenannte Assoziationsliste erkennen [(–)]. Der car dieser Liste ist der Schlüssel, der cadr ist der zugehörige Wert. Diesen erhalten wir mit der Funktion **assoc**. Abb. 3 zeigt einige Zugriffe auf Elemente des Rahmens mit Hilfe der Funktion **assoc**.

```
XLISP version 1.72, (c) 1986, by David Betz, converted for MEGAMAX by Karl Sarnw
; loading "PP.lsp"
; loading "TRACE.lsp"
; loading "D:\LISP\FRAME.LSP"
> (get 'atari 'rahmen)
(ATARI (MODELL (WERT 1040ST)) (SPEICHER (STANDARD 1MB) (WERT 2MB)) (MONITOR (WE)
> (pp (get 'atari 'rahmen))
(ATARI (MODELL (WERT 1040ST))
       (SPEICHER (STANDARD 1MB) (WERT 2MB))
       (MONITOR (WERT COLOR) (STANDARD SW)) )
T
> (assoc 'speicher (get 'atari 'rahmen))
(SPEICHER (STANDARD 1MB) (WERT 2MB))
> (assoc 'wert (assoc 'speicher (get 'atari 'rahmen)))
(WERT 2MB)
> (cadr (assoc 'wert (assoc 'speicher (get 'atari 'rahmen))))
2MB
> █
```

Abb.3: Zugriff auf Elemente eines Rahmens auf der Propertyliste.

Die assoc-wurschtelei läßt sich natürlich durch Definition einer Funktion umgehen, die den Namen des Rahmens, Schlitzes und Merkmals als Parameter übernimmt und die Liste mit dem Merkmalswert zurückgibt. Diese Funktion erhält den sinnvollen Namen **rhol** und findet sich zu Beginn des Listing1. Die Funktion **rhol-rahmen** gibt den ganzen Rahmen zurück. Falls der gewünschte Rahmen allerdings nicht vorhanden ist, wird er mit Hilfe der setf Funktion neu erzeugt.

Das Erzeugen eines Rahmens in der Propertyliste

Diese Funktion ist erheblich komplizierter zu realisieren und bedarf daher einiger Vorarbeit. Zunächst einmal muß getestet werden, ob der Rahmen exisitert. Wenn dies nicht der Fall ist, wird der Rahmen mit der obigen Funktion 2 rhol-rahmen erzeugt. Andernfalls wird der Rahmen mit derselben Funktion als Assoziationsliste geholt und auf das Vorliegen eines entsprechenden Merkmals getestet. Es versteht sich, daß derselbe Wert nicht zweimal abgespeichert wird. Liegt der Wert des Merkmals also bereits vor, dann wird nicht noch einmal gespeichert, sondern einfach **NIL** zurückgegeben. Besondere Beachtung sollte hierbei die Art und Weise der Erweiterung finden, wenn ein Schlitz, bzw. ein Merkmal noch nicht vorliegen. Abb. 4 zeigt die Vorgehensweise, die wir uns nun etwas näher unter die Lupe nehmen. Zu Beginn soll z.B. der Rahmen **(ATARI (MODELL (WERT 1040ST)))** auf der Propertyliste vorliegen und es soll der Schlitz **SPEICHER** mit dem Merkmal **(WERT 2MB)** hinzugefügt werden. Dazu rufen wir die Funktion **(rpack 'atari 'speicher 'wert '2mb)** auf. Der interessanteste Teil der Funktion läuft bereits bei der Initialisierung mit dem LET-Konstrukt ab. Hier wird **wert-liste** als gebundene Variable erzeugt, mit einem Wert, der sich durch Aufruf der Funktion

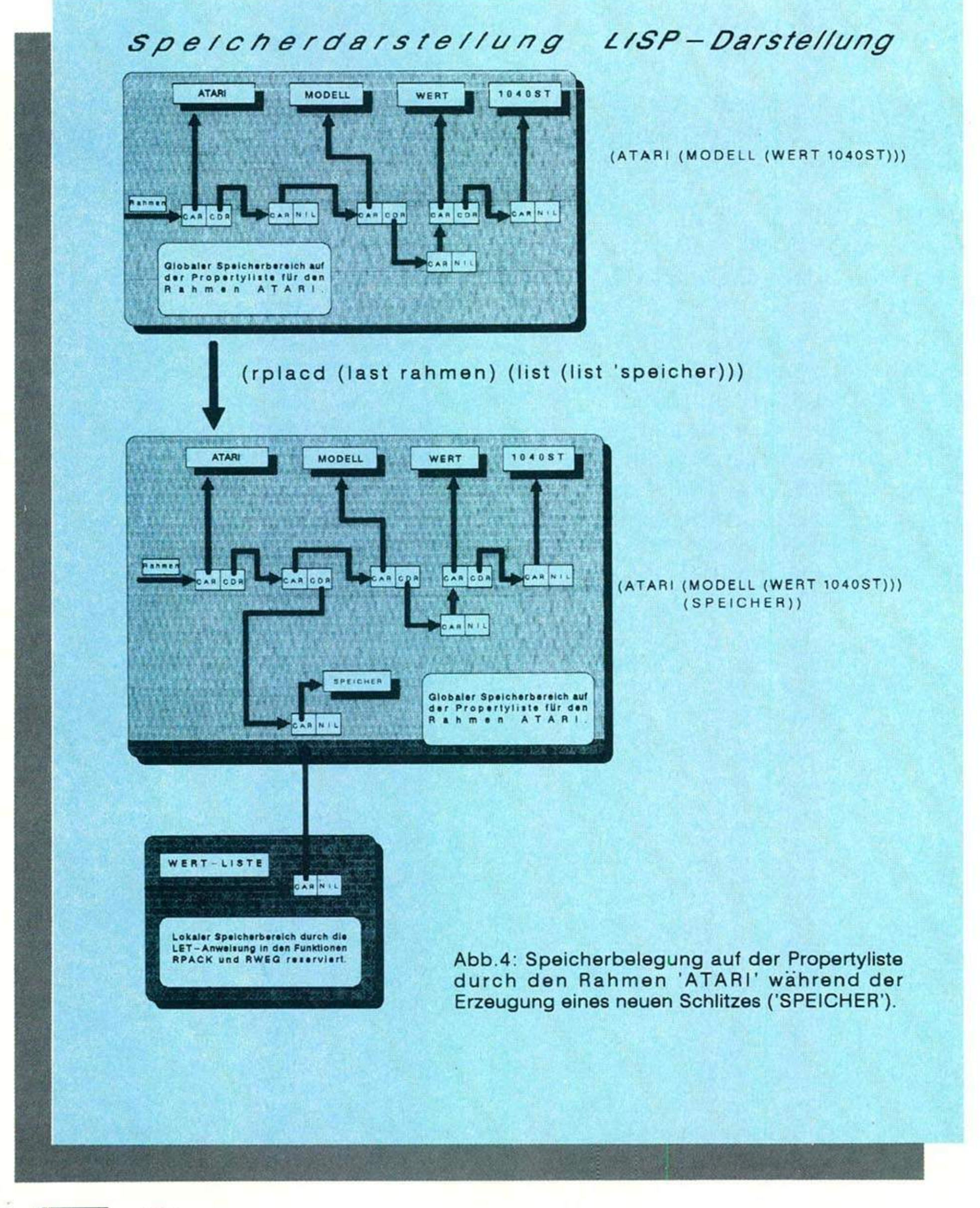


Abb.4: Speicherbelegung auf der Propertyliste durch den Rahmen 'ATARI' während der Erzeugung eines neuen Schlitzes ('SPEICHER').

(folge-pfad '(speicher wert) '(atari (modell (wert 1040st)))) ergibt (siehe Listing 1). Diese Funktion testet, ob die Liste mit dem Pfad (d.h. **'(speicher wert))** bereits leer ist. Dann wird die Argumentliste liste zurückgegeben. Sonst ruft sich die Funktion selbst auf, allerdings mit dem Rest des Pfades und dem Ergebnis der Funktion **(erweitern 'speicher '(atari (modell (wert 1040st)))**. In dieser Funktion erfolgt dann der eigentliche Teil der Erweiterung des Rahmens. Die erste Klausel der Disjunktion trifft zu, wenn der Schlüssel **(SPEICHER)** bereits in der Assoziationsliste des Rahmens enthalten ist. Dann braucht natürlich nicht erweitert zu werden, und die Funktion gibt die Assoziationsliste mit dem Schlüssel und dem Wert zurück, um dann im nächsten Schritt von folge-pfad als Ergebnis zurückgegeben zu werden. Falls nun aber wie in unserem Beispiel der Schlüssel **SPEICHER** noch nicht in der Assoziationsliste des Rahmens enthalten ist, tritt die zweite Klausel der Disjunktion in der Funktion **erweitern** in Kraft. Diese ersetzt den CDR des letzten Listenknotens durch die Liste die die Liste des Schlüssels ersetzt (mittlerer Teil der Abb. 4). Damit wurde an die Liste etwas angefügt, die ursprüngliche Liste also verändert. An diesem Beispiel sollte klar geworden sein, daß LISP-Knoten nichts anderes als Knoten eines binären Baumes sind, deren erster Teil ein Zeiger auf ein Datenelement ist und dessen zweiter Teil ein Zeiger auf den nächsten LISP Knoten. Als Funktionswert wird dann die Liste mit dem Schlüssel (d. h. dem Schlitz **SPEICHER)** zurückgegeben. Diese wird dann beim Rücksprung aus der tieferen Rekursionstufe an die lokal gebundene **wert-liste** übergeben. Für Anfänger verblüffend ist dabei wahrscheinlich, daß trotz Verwendung einer lokal gebundenen Variablen (der Liste **wert-liste)** ein globaler Effekt erzielt wird. Der untere Teil der Abb.4 illustriert diesen Effekt. In der Funktion 2 rpack wird nämlich (wenn der Wert **2MB** nicht in der Wertliste ist, wie bei uns der Fall!) der CDR des letzten Elementes der Werteliste durch die Liste mit dem Wert **2MB** ersetzt. Da aber der CAR des LISP-Knotens von 2 wert-liste auf einen global gebundenen LISP-Knoten zeigt, wird beim Anfügen eines Elementes an die Liste von **wert-liste** gleichzeitig ein globaler Effekt erreicht und der CDR des letzten Elementes des Rahmens hinzugefügt.

Dämonen als stille Helfer

Jeder Programmierer kennt die Situation: Man fragt einen Datensatz ab und stellt fest, daß die abgefragte Information nicht vorhanden ist. Da wünscht man sich schon manchmal ein paar Dämonen, die ohne viel Aufsehen die passenden Werte besorgen, damit die anfragende Routine nicht abstürzt oder falsche Ergebnisse produziert. Solche Dämonen aber kann man in einem Rahmen leicht implementieren. Schauen wir uns zu diesem Zweck einmal die Datenbank über Automobile in Listing 1 an. Dort sehen wir, daß die Informatioen über die verschiedenen Autotypen nicht bei allen Typen vollständig sind. Über VW weiß die Datenbank z.B. nur, daß der meistgekauft Typ der GOLF ist, weshalb er als Standard eingesetzt wurde. Die Anfrage **(rhol 'vw 'typ 'wert)** würde also NIL ergeben, da das gesuchte Merkmal im Schlitz **typ** nicht vorhanden ist. Dagegen liefert die Anfrage 2 (rhol-w-s 'vw 'typ) erwartungsgemäß **(GOLF)**, da nach mißlungener Abfrage nach dem Wert, die Abfrage nach dem Merkmal Standard im gleichen Schlitz erfolgt. Ein Dämon kann aber noch mehr. Wenn wir das Alter eines Autos nicht kennen, dann könnten wir einen **WENN-NOETIG** Dämon aktivieren, der einspringt, wenn weder ein Standard noch ein Wert vorliegen. Nehmen wir einmal an, wir wollten für eine Gebrauchtwagendatenbank den Vorbesitzer eines Wagens in den Rahmen mit aufnehmen. Dazu verpassen wir z.B. dem Rahmen **OPEL** den **WENN-NOETIG** Dämon **Frage** für den Schlitz Vorbesitzer:

(rpack 'opel 'vorbesitzer 'wenn-noetig 'frage)
(FRAGE)
Fragen wir nun nach dem Vorbesitzer des Opel:
(rhol-w-s-d 'opel 'vorbesitzer)

Da weder ein Wert für den Vorbesitzerschlitz, noch ein Standard bekannt ist, wird der Dämon aktiv (wir haben rhol-w-s-d aufgerufen!) und läßt die Funktion Frage ablaufen, die eine Eingabe für den Wert verlangt und diesen im Schlitz abspeichert. Abb. 5 zeigt das Protokoll der Abfrage.

```
XLISP version 1.72, (c) 1986, by David Betz, converted for MEGAMAX by Karl Sarnw
; loading "PP.lsp"
; loading "TRACE.lsp"
; loading "D:\LISP\FRAME.LSP"
> (rpack 'opel 'vorbesitzer 'wenn-noetig 'frage)
FRAGE
> (rhol-w-s-d 'opel 'vorbesitzer)
(BITTE GEBEN SIE EINEN WERT FUER SCHLITZ < VORBESITZER > IN DEM RAHMEN < OPEL >)

> georg
(GEORG)
> (pp (rhol-rahmen 'opel))
(OPEL (IST (WERT AUTO))
      (TYP (STANDARD REKORD))
      (VORBESITZER (WENN-NOETIG FRAGE)
                   (WERT GEORG) ) )
T
>
```

Abb.5: Protokoll der heimlichen Mitarbeit eines Daemons.

Noch eindrucksvoller ist allerdings die Vererbung von Eigenschaften. Um einen Dämon zu implementieren müßten wir nun alle Automobile mit dem entsprechenden **wenn-noetig** Dämon versehen. Viel einfacher wäre es, könnte man für alle Autos gleichzeitig einen entsprechenden Dämon vereinbaren. Das ist möglich, wenn man einen Schlitz **Ist** vereinbart, der die Art des Rahmens definiert. Also z.B. **(rpack 'lancia 'ist 'wert 'auto)**. Weiterhin führen wir umgekehrt einen Rahmen **Auto**, der ebenfalls einen Schlitz **Ist** enthält und in dem als Merkmal **Wert** alle Autos der Datenbank aufgeführt sind (siehe Listing 1). Die Funktionen **rhol-i, rhol-n** und **rhol-z** holen sich über den Schlitz **Ist** Informationen der verwandten Rahmen und vererben diese Informationen weiter. Die vererbten Informationen können Werte aber auch Dämonen sein. Abb. 6 zeigt die Abfrage

```
XLISP version 1.72, (c) 1986, by David Betz, converted for MEGAMAX by Karl Sarnw
; loading "PP.lsp"
; loading "TRACE.lsp"
; loading "D:\LISP\FRAME.LSP"
> (rhol-w-s-d 'vw 'alter)
NIL
> (rhol-i 'vw 'alter)
(1)
> (rhol-n 'vw 'alter)
(1)
> (rhol-z 'vw 'alter)
(BITTE GEBEN SIE EINEN WERT FUER SCHLITZ < ALTER > IN DEM RAHMEN < AUTO > EIN)

> 3
(3)
> █
```

Abb.6: Daemonen koennen vererbt werden.

d 'vw 'alter). Da kein Schlitz **ALTER** in der Assoziationsliste des Rahmens enthalten ist, liefert weder die Frage nach dem Merkmal **WERT**, noch die Frage nach dem Merkmal **STANDARD** noch die Anfrage nach dem Merkmal **WENN-NOETIG** (Dämon) einen Wert.
Die nächste Anfrage ist **(rhol-i 'vw 'alter)**. Diesmal wird die wurde, wird der Merkmalswert zurückgegeben. Aus diesem Grunde ist die Antwort in Abb. 6 auch (1), da im Rahmen **LANCIA** im Schlitz **ALTER** im Merkmal **WERT** der Wert 1 gefunden wurde.
Das Ergebnis der Abfrage **(rhol-n 'vw 'alter)** ist das gleiche, weil in der ersten Klausel der Hilfsfunktion rhol-n1 ebenfalls alle Elemente der Liste **Klassen** auf das Vorliegen des Merkmals **WERT** getestet werden.

unter Benutzung der Vererbung.

Vererbung von Informationen

Die drei Funktionen **rhol-i, rhol-n** und **rhol-z** übernehmen die Vererbung von Informationen. Alle Vererbung geschieht dabei über den 2 Ist-Schlitz. Schauen wir uns dazu einmal Abb.6 genauer an. Die erste Abfrage lautet **(rhol-w-s-**

Informationsvererbung über den **Ist-Schlitz** des Rahmens verwendet. Dazu werden mit Hilfe der Funktion **rhol-klassen** alle durch den **Ist1 - Schlitz miteinander verbundenen Rahmen aufgefunden. Die Hilfsfunktion 2 rhol-i1** schaut nun in jedem Rahmen der Liste **Klassen** nach, ob der gefragte Schlitz (in unserem Beispiel **ALTER**) mit dem Merkmal **WERT** vorhanden ist. Wenn ein Rahmen mit diesem Merkmal gefunden

Anders reagiert das Programm lediglich auf die Anfrage **(rhol-z 'vw 'alter).** Nunmehr werden nämlich alle Elemente der Liste **Klassen** zunächst auf das Vorliegen der Merkmale **WERT, STANDARD** und **WENN-NOETIG** (Dämon) untersucht, bevor das nächste Element der Klassen untersucht wird. Und als Dämon findet **rhol-z** in der Assoziationsliste des Rahmens **AUTO** (steht an erster Stelle der Liste **Klassen)** das Merkmal **WENN-NOETIG** mit dem Merkmal **FRAGE**. Der Dämon **WENN-NOETIG** wird also aktiv und fragt nach dem Alter. Abb. 7 zeigt die Art der Vererbung, wie sie durch **rhol-i, rhol-n1 und rhol-z** verwendet wird. Man erkennt, daß die Buchstaben mnemonische Bedeutung haben. Die Reihenfolge der Abarbeitung ist allerdings willkürlich gewählt (Wo bliebe sonst die mnemonische Wirkung?).

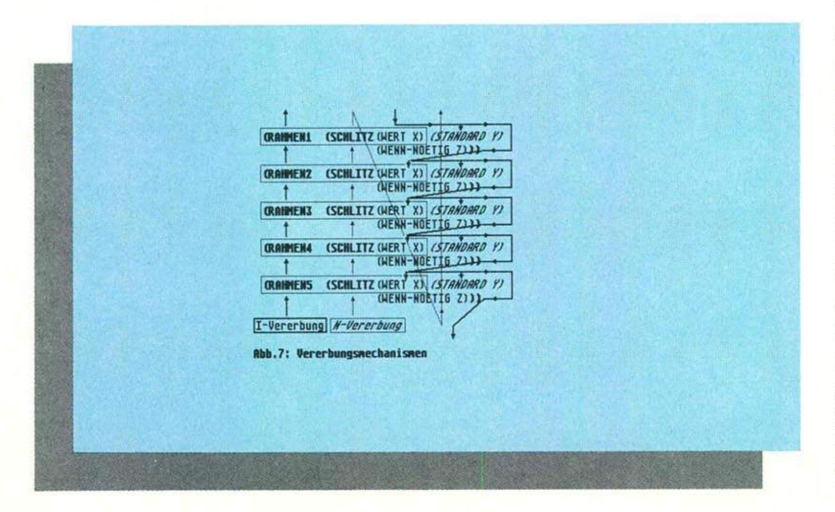


Abb.7: Vererbungsmechanismen

P R O G R A M M E



Megamax Modula-2

Modula-2 Entwicklungssystem mit GEM-Shell.
398,- DM

„Bolo", die Demospieldiskette oder Demodisk für 20,- DM



Megamax-C

Das bekannte C-Entwicklungssystem. Fragen Sie nach den Ergänzungstools.

Neuer Preis: 398.- DM



Signum!Zwei

Die zweite Generation der umfassenden Textverarbeitung für unterschiedlichste Anwendung. Für Literaten und Musiker, für Sprachwissenschaftler, Physiker, Chemiker, Mathematiker usw.
Viele zusätzliche Fonts sind erhältlich.

448,- DM

gegen die Tücken



STAD

Das Grafik-Programm, das fast keine Wünsche offen läßt.
Die neue Version: 99 Bildschirme beim Mega-ST.
Laserdruckertreiber für Atari-Laser.

179,- DM



Creator

Grafik gekonnt in Bewegung gesetzt. Zeichnen, Animation, Zeichentrickfilm. Das Zeichenprogramm, das Bilder bewegt. Und noch ein bißchen mehr.

Demo gegen 20,- DM



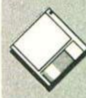
Imagic

Der Grafik-Compiler.
Die Sprache, die Bilder schreibt.

498,- DM

des Objekts:

Utility Series: Eine neue Reihe nützlicher (weil heilsamer) „kleiner" Hilfsprogramme in sparsamer Verpackung zu kulantem Preis. Die ersten beiden:



FlexDisk

Die flexible Ramdisk für den blitzschnellen Zugriff auf alle Daten. Bootfähigkeit ermöglicht Accessories und Autoordner auf Ramdisk.
Info anfordern.

69,- DM



Das Harddisk Utility

Sicherheitskopien von der Harddisk jetzt ohne Umstände. Harddisk-Backup Programm mit einfachster Bedienung: Schnell, sicher und GEM eingebunden. Ein Muß für Profis. Info anfordern.

69,- DM

Editor Toolbox

Programm-Editor mit vielen Features sowie vollständig dokumentierter C-Source für Megamax-C.

149,- DM

Gegen jedes Unkraut sei ein Mittel gewachsen – sagt der Volksmund. Wir dagegen haben Mittelchen gegen die verschiedensten Symptome, mit denen sich ein fleißiger ST-Anwender konfrontiert sehen könnte, denn ST-Anwender sind unsere Freunde (außer vielleicht Herbert aus Remscheid)*.

Da wäre die Stummheit der Maschine und da sind die Sprachen. Hier der Zettelwust und da die Textverarbeitung. Last not least die Grafik- und Zeichenprogramme, anzuwenden bei akuten Kreativitätsanfällen oder schlicht: bei Langeweile.

Frohe Feiertage und einen guten Rutsch wünscht Application Systems Heidelberg

** Name und Wohnort wurden aus Klanggründen vom Texter geändert!*

APPLICATION SYSTEMS HEIDELBERG, Englerstraße 3, Postfach 102 646, D-6900 Heidelberg, Telefon (0 62 21) 30 00 02. **VERTRIEB, Benelux:** PDS, Promenadeplein 107, NL-2711 AB Zoetermeer. **Österreich:** Ueberreuter Handels- und Verlags GmbH, Laudongasse 29, A-1082 Wien. **Schweiz:** Senn Computer AG, Langstraße 31, CH-8021 Zürich. **Nordeuropa:** New World Computer/Atari Danmark, Skanderborgwej 14, DK-8260 Viby J. **Frankreich:** APPLICATION SYSTEMS PARIS, 12, rue Edouard Jacques, F-75014 Paris. **USA:** APPLICATION SYSTEMS DALLAS.



MMAD

XLISP's Geisterstunde (Dämonenaktivierung)

Wie aktiviert man nun einen Dämon? Das ist in der Tat ein Paradestück für die Leistungsfähigkeit von LISP. Schließlich muß der Interpreter nun je nach Dämon ein anderes LISP Programm aktivieren, daß irgendwo als Liste vorliegt. Den Code für die Dämonenaktivierung finden wir in den Funktionen **rhol-w-s-d** und 2 rhol-n2. In beiden findet man die Funktion **mapcan** als Motor der Aktivierung. Die Funktion **mapcan** hat folgende Syntax:

(mapcan <funktion> <liste1> [<liste2>...])
Es bedeuten:
<funktion>: Eine gültige LISP Funktionsdefinition.
<liste?>: Für jedes Argument der Funktion <funktion> eine Liste mit Argumentwerten.

Die Funktion wendet nun <funktion> auf jedes Element der Listen an. Die zurückgegebenen Werte werden zu einer Liste zusammengefaßt. Leider ist **mapcan** in der XLISP Dokumentation nicht beschrieben, funktioniert aber trotzdem. Die obige Definition stimmt mit der Definition von **mapcar** in der XLISP Dokumentation überein. Trotzdem arbeiten beide Funktionen leicht unterschiedlich. Während **mapcar** die Ergebnisse in eine Liste packt, zieht **mapcan** die Ergebnisse in einer Liste mittels **nconc** zusammen. Das setzt natürlich voraus, daß das Ergebnis der Funktion <funktion> eine Liste ergibt, da sonst nicht zusammengezogen werden kann.

Beispiel:
(mapcar 'equal '(1 2 3) '(2 2 2))
(NIL T NIL)

(mapcar '(lambda (a b) (list (equal a b))) '(1 2 3) '(2 2 2))
((NIL) (T) (NIL))

(mapcan '(lambda (a b) (list (equal a b))) '(1 2 3) '(2 2 2))
(NIL T NIL)

Mapcar und **mapcan** erlauben also die Anwendung der gleichen Funktion auf mehrere Parameter hintereinander. Im Falle unserer Dämonen, holt sich die Funktion **rhol-w-s-d** also zunächst mit **(rhol rahmen schlitz 'wenn-noetig)** Liste aller als Dämon agierender Funktionsnamen und wendet dann die Schablone im **mapcan** Aufruf sukzessiv an.

Vererbung in Analogieschlüssen

Wie Eingangs erwähnt, ist analoges Schließen ein typisches Merkmal von Intelligenz. Aus 2 stammt das folgende Beispiel. Wenn Jemand sagt, **Fred ist wie ein Bär**, dann überträgt der Zuhörer Eigenschaften des Bärs auf den ihm unbekannten Fred. Ein Anlogieschluß ist also nichts anderes als das Übertragen (Vererben) von Informationen. Und da wir im letzten Abschnitt mehrere Methoden zur Informationsvererbung kennengelernt haben, soll das oben genannte Beispiel in unserer Rahmensprache nachvollzogen werden. Im letzten Teil des Listing 1 finden Sie die Rahmen für Fred und Ted. Abb. zeigt, wie die Größe und Gangart von Fred aus der Analogie zu Ted vererbt wird.

Schlußwort

Diese Serie ist am Ende angelangt. Es gäbe natürlich noch vieles zu berichten, vor allem, weil der Fortschritt im Bereich der Informatik so ungeheuer schnell voranschreitet. So habe ich beispielsweise gerade vor ein paar Monaten von den ersten Neuronenchips erfahren, die in den USA verkauft werden. Parallel verarbeitende Rechnerarchitekturen werden in Zukunft sicherlich eine bedeutende Rolle spielen. Kurz und gut, wir stehen vor der Schwelle vieler neuer Verfahren und Methoden und vieles (wenn auch sicher nicht alles) was heute kaum vorstellbar ist, wird sich mit neuen, von der herkömmlichen v. Neumann Architektur abweichenden Konzepten erreichen lassen. Ich bedanke mich bei Ihnen, lieber Leser, daß Sie mir bis hierher gefolgt sind. Als kleines Dankeschön füge ich der PD demnächst eine Shell für XLISP und TOY-Prolog bei, sodaß der Benutzer sehr bequem einen Entwicklungszyklus mit Hilfe eines Editors aus der PD durchführen kann.

```
XLISP version 1.72, (c) 1986, by David Betz, converted for MEGAMAX by Karl Sarnw
; loading "PP.lsp"
; loading "TRACE.lsp"
; loading "D:\LISP\FRAME.LSP"
> (rhol-i 'fred 'groesse)
NIL
> (rhol-n 'fred 'groesse)
(GROSS)
> (rhol-z 'fred 'gangart)
(POLTERND)
> █
```

Abb.8: Analogieschluss durch Vererbung von Information

Rahmen-implementation nach Winston und Horn

```
1:   ;Rahmenimplementation nach Winston und Horn:
     LISP. Second Edition 1984.
2:   ;********************************
3:
4:   (defun rhol (rahmen schlitz merkmal)
5:           (cdr (assoc merkmal (cdr (assoc
             schlitz (cdr (get rahmen
             'rahmen)))))))
6:
7:   ;Holt den Wert eines Rahmens aus einem
      Schlitz mit einem bestimmten Merkmal
8:   ;********************************
9:
10:  (defun rhol-rahmen (rahmen)
11:          (cond ((get rahmen 'rahmen))
12:                (t (setf (get rahmen 'rahmen)
                   (list rahmen)))))
13:
14:  ;Holt die ganze Assoziationsliste eines
      Rahmens von der Propertyliste
15:  ;********************************
16:
17:  (defun erweitern (schluessel liste)
18:          (or (assoc schluessel (cdr liste))
19:              (cadr (rplacd (last liste)
20:              (list (list schluessel))))))
21:
22:  ;Erweitert die Assotiationsliste des Rahmens
      um einen Schlüssel, wenn er noch nicht in
23:  ;der Assoziationsliste vorhanden ist.
24:  ;********************************
25:
26:  (defun folge-pfad (pfad liste)
27:          (cond ((null pfad) liste)
28:              (t (folge-pfad (cdr pfad)
                 (erweitern (car pfad) liste)))))
29:
30:  ;Gibt die Liste mit dem Schlitz und Merkmal
      zurück, falls es schon in der
31:  ;Assoziationsliste des Rahmens existiert.
      Sonst nur die Liste mit dem neu
32:  ;eingetragenen Schlüssel.
33:  ;********************************
34:
35:  (defun rpack (rahmen schlitz merkmal wert)
36:          (let ((wert-liste (folge-pfad
             (list schlitz merkmal)
37:              (rhol-rahmen rahmen))))
38:              (cond ((member wert wert-liste)
                 nil)
39:                    (t (rplacd (last wert-list
                       e) (list wert)) wert))))
40:
41:  ;Erzeugt einen neuen Rahmen, Schlitz oder
      Merkmalswert, falls noch nicht auf der
42:  ;Propertyliste vorhanden.
43:  ;********************************
44:
45:  (defun rweg (rahmen schlitz merkmal wert)
46:          (let ((wert-liste (folge-pfad (list
             schlitz merkmal)
47:              (rhol-rahmen rahmen))))
48:                (cond ((member wert wert-liste)
49:                      (delete wert wert-liste) t)
50:                         (t nil))))
51:
52:  ;vernichtet einen Rahmen.
53:  ;********************************
54:
55:  (defun rcheck (rahmen schlitz merkmal wert)
56:          (cond ((member wert (rhol rahmen
             schlitz merkmal)) t)
57:                (t nil)))
58:
59:  ;Checkt eine Rahmen-Schlitz-Merkmal-Wert-
     Kombination auf Existenz.
60:  ;********************************
61:
62:  (defun rklammer (rahmen1 rahmen2 schlitz)
63:          (rplacd (rhol-rahmen rahmen1)
64:          (list (folge-pfad (list schlitz)
             (rhol-rahmen rahmen2))))
65:          schlitz)
66:
67:  ;Klammert zwei Rahmen bei einem bestimmten
      Schlitz zusammen
68:  ;(Zwei Rahmen erben gleiche Merkmale).
69:  ;********************************
70:
71:  (defun rhol-klassen (start)
72:    (reverse (rhol-klassen1 (list start)nil)))
73:
74:  (defun rhol-klassen1 (liste klassen)
75:          (cond ((null liste) klassen)
76:              ((member (car liste) klassen)
              (rhol-klassen1 (cdr liste)
              klassen))
77:                (t (rhol-klassen1 (append (rhol
                 (car liste) 'ist 'wert) (cdr
                 liste))
78:                  (cons (car liste) klassen)))))
79:
80:  ;Die beiden Funktionen liefern eine Liste
      aller durch den Schlitz "IST"
81:  ;miteinander verbundnen Rahmen.
82:  ;********************************
83:
84:  (defun rhol-z (rahmen schlitz)
85:     (rhol-z1 schlitz (rhol-klassen rahmen)))
86:
87:
88:  (defun rhol-z1 (schlitz klassen)
89:          (cond ((null klassen) nil)
90:                ((rhol-w-s-d (car klassen)
                 schlitz))
91:                (t (rhol-z1 schlitz
                 (cdr klassen)))))
92:
93:  ;Gibt den Wert eines Rahmen-Schlitzes zurück,
94:  ;indem zunächst das Merkmal "WERT" gesucht
     wird. Fehlt dieses,
95:  ;wird das Merkmal "STANDARD" gesucht. Fehlt
      auch dieses,
96:  ;wird das Merkmal "WENN-NOETIG" nach Dämonen
      abgesucht.
97:  ;********************************
98:
99:  (defun rhol-n (rahmen schlitz)
100:    (let ((klassen (rhol-klassen rahmen)))
101:      (cond ((rhol-n1 schlitz klassen 'wert))
102:                ((rhol-n1 schlitz klassen
                   'standard))
103:                ((rhol-n2 schlitz klassen
                   'wenn-noetig))
104:                (t nil))))
105:
106:
107: (defun rhol-n1 (schlitz klassen schluessel)
108:    (cond ((null klassen) nil)
109:        ((rhol (car klassen) schlitz
          schluessel))
110:           (t (rhol-n1 schlitz (cdr klassen)
           schluessel))))
111:
112:
113: (defun rhol-n2 (schlitz klassen schluessel)
114:         (cond ((null klassen) nil)
115:               ((mapcan (lambda (demon)
                 (funcall demon (car klassen)
                 schlitz))
116:                       (rhol (car klassen)
                         schlitz schluessel)))
117:               (t (rhol-n2 schlitz (cdr
                klassen) schluessel))))
118:
119: ;Wie rhol-z, allerdings werden sämtliche
     Elemente der Liste Klassen
120: ;zuerst auf Werte, Standard und Wenn-noetig
     Merkmale untersucht.
121: ;****************************************
122:
123: (defun rhol-i (rahmen schlitz)
124:    (rhol-i1 (rhol-klassen rahmen) schlitz))
125:
126:
127: (defun rhol-i1 (klassen schlitz)
128:    (cond ((null klassen) nil)
129:        ((rhol (car klassen) schlitz 'wert))
130:           (t (rhol-i1 (cdr klassen)
            schlitz))))
131:
132: ;Wie rhol-z, allerdings schaut diese Funktion
      nur nach Werten.
133: ;****************************************
134:
135:
136: (defun rhol-w-s (rahmen schlitz)
137:         (cond ((rhol rahmen schlitz 'wert))
138:               ((rhol rahmen schlitz
                 'standard))))
139:
140: ;Hole Wert eines Rahmen-Schlitz-Merkmals.
      Wenn nicht vorhanden schaue
141: ;bei Standard nach.
142: ;********************************
143:
144:
145: (defun rhol-w-s-d (rahmen schlitz)
146:    (cond ((rhol-w-s rahmen schlitz))
147:        (t (mapcan (lambda (demon)(funcall
          demon rahmen schlitz))
```

TiM

Eine **Buchführung** für den **Atari ST** (Monochrom)

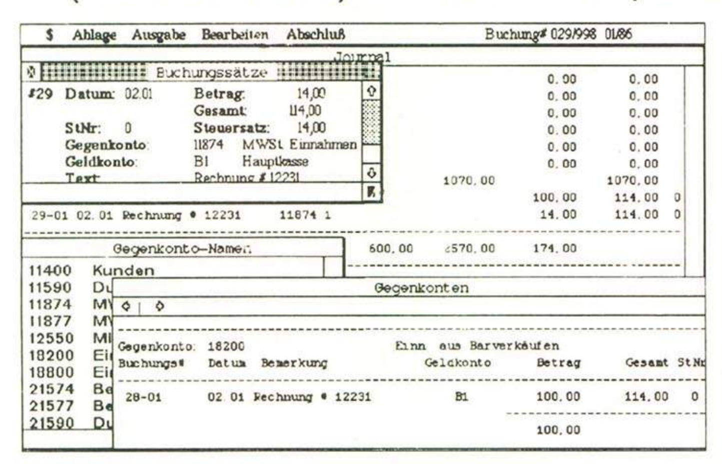


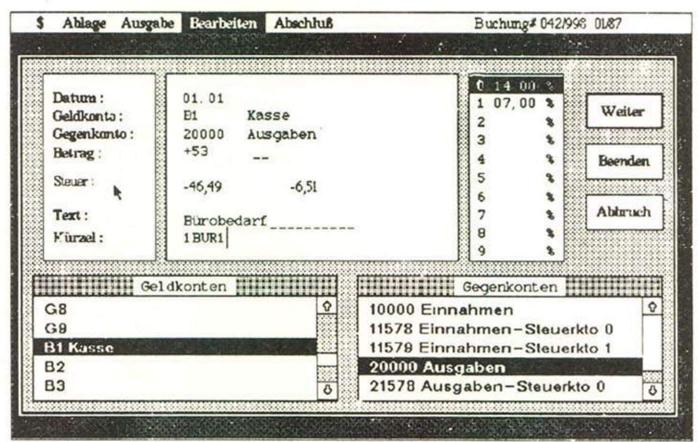
Software

C. A. $. H. GmbH

NEU

Version 1.1





Wenn Sie wissen wollen, wie unsere Software in der Fachpresse beurteilt wird, lesen Sie die Tests über TiM 1.0 in

- ***c't 5/87***
- ***68000er 7/87***
- ***ST Computer 8/87***

und

- ***Atari Magazin 5/87***

Inzwischen haben wir unsere Buchführung TiM um Vieles erweitert und zum Kernstück unserer **Time is Money** Serie gemacht, die wir Ihnen über die kommenden Monate vorstellen werden.

Buchführung TiM 1.1 besitzt: **5**-stellige Konten und Kürzel, **10** frei definierbare Steuersätze, **27** (3X9) Geldkonten, **2000** Gegenkonten, **998** Buchungen/Periode, Ausgabe der Umsatzsteuer, Mitführen der Bruttobeträge, Druckerinitialisierung, Drucker mit **80 oder 96** Zeichen/Zeile, Buchen mit Maus + Tastatur, Ausgabe an Bildschirm, Drucker oder Datei. Und eine Bildschirmdarstellung die dem Atari ST gerecht wird, wie Sie oben sehen können.

Wenn Sie Ihre Buchführung selbst erledigen wollen, oder nur einen Überblick über Einnahmen und Ausgaben benötigen, um dann Ihren Steuerberater aufzusuchen, ist TiM für Sie geeignet.

TiM 1.1 ist nicht kopiergeschützt und wird mit 80-seitigem Handbuch geliefert. Drucker wird empfohlen.

TiM 1.1 erhalten Sie im Fachhandel oder direkt bei uns.

TiM 1.1	**DM**	**298,--***
Handbuch	**DM**	**30,--*†**
TiM-Demo	**DM**	**10,--***

* unverbindliche Preisempfehlung
† wird bei Kauf angerechnet, falls Direktbestellung vorliegt.

TiM 1.1 erhalten Sie gegen Vorauskasse oder NN, alles andere **nur** gegen Vorauskasse.
Bei Nachnahmebestellung zuzüglich DM 10,-- Versandkosten.

Sie erhalten unsere Time is Money-Serie, Demos und Prospektmaterial auch bei folgenden Händlern:

PLZ Ort	Händler
1100 A/Wien	Vogler
5402 CH/Baden	A-Magic Computer
1000 Berlin	Dataplay
1000 Berlin	Digital Computer
1000 Berlin	BH Garten des Wissens
1000 Berlin	Karstadt
1000 Berlin	Schäfer
2000 Hamburg	Createam Computer
2000 Hamburg	Kabs & Winterscheit
2000 Hamburg	Ing.-Büro Kludasch
2300 Kiel	MicroComputer Christ
2350 Neumünster	SCM Computer Magazin
2800 Bremen	PS-Data
2940 Wilhelmshaven	Radio Tiemann
3000 Hannover	Datalogic
3170 Gifhorn	Computer-Haus Gifhorn
3500 Kassel	Hermann Fischer GmbH
4000 Düsseldorf	Borho + Partner
4000 Düsseldorf	Rainer Driesen
4200 Oberhausen	LaSch GbR
4300 Essen	Karstadt
4350 Recklinghausen	Computer Centrale
4410 Warendorf	Digital Works
4500 Osnabrück	Heinecke Electronic
4531 Lotte/Büren	Bruns Computerorganisation
4600 Dortmund	Knupe
4650 Gelsenkirchen	ComTron
4800 Bielefeld	CSF Computer
4800 Bielefeld	Microtec
5000 Köln	Büromaschinen Braun
5000 Köln	Callsoft Koeln
5010 Bergheim	Computerstudio Hölscher
5060 Berg. Gladbach	HSS GbR
5200 Siegbu g	Computer Center
5540 Prüm	ATC Computer
5630 Remscheid	C O M Soft
5800 Hagen	Axel Böckem
5860 Iserlohn	Compuvet, Dr. Sasum
6000 Frankfurt	BCO
6000 Frankfurt	Schmitt Computersysteme
6094 Bischofsheim	Grosse Datensysteme
6300 Giessen	Interface
6330 Wetzlar	Computer Fachmarkt
6340 Königstein	KFC Computersysteme
6500 Mainz	Schmitt Computersysteme
6600 Saarbrücken	Kaufhof AG
6700 Ludwigshafen	MKV GmbH
6720 Speyer	MKV GmbH
6750 Kaiserslautern	Gotthold Bürocenter
7000 Stuttgart	BNT
7100 Heilbronn	Fritz Seel
7100 Heilbronn	Walliser & Co.
7410 Reutlingen	Computershop Brock
7475 Meßstetten	Scheurer Computermarkt
7500 Karlsruhe	Papierhaus Erhardt
7520 Bruchsal	Helmut Jöst
7700 Singen	Udo Meier
7940 Riedlingen	Schlegel Datentechnik
7980 Ravensburg	Grahle
8000 München	Ludwig Computer
8000 München	Philgerma GmbH
8000 München	Schulz Computer
8150 Holzkirchen	Münzenloher GmbH
8170 Bad Tölz	Elektronik Center
8400 Regensburg	Elektroland Zimmermann
8500 Nürnberg	Schmitt Computersysteme
8900 Augsburg	Adolf & Schmoil
8900 Augsburg	Schmitt Computersysteme
8960 Kempten	Staehlin

Achtung! Update von TiM 1.0 auf TiM 1.1: Schicken Sie uns Ihre Originaldiskette TiM 1.0, Registrierkarte und selbstadressierten mit DM 2,50 freigemachten C5 Umschlag. Sie erhalten TiM 1.1 mit Handbuch und Updateprogramm kostenlos zugesandt.

Generalvertrieb Schweiz:
A-Magic Computer
P.O. Box 2065, CH-5402 Baden
Telefon 071 71 45 82

C. A. $. H. GmbH
Schillerstr. 64, 8900 Augsburg
Telefon 0 82 37 / 10 20

Listings (Fortsetzung)

```
148:            (rhol rahmen schlitz
                 'wenn-noetig)))))
149:
150:  ;Wie rhol-w-s, schaut aber zusätzlich noch
       nach Wenn-nötig Dämonen, wenn
151:  ;weder ein Wert noch ein Standard vorliegt.
152:  ;***********************************
153:
154:
155:  (defun frage (rahmen schlitz)
156:          (print `(bitte geben Sie einen wert
                fuer schlitz < ,schlitz > in dem
                rahmen < ,rahmen > ein))
157:          (terpri)
158:          (let ((response (read)))
159:                  (cond (response (rpack rahmen
                          schlitz 'wert response)
                           (list response))
160:                        (t nil))))
161:
162:  ;Beispiel eines Wenn-nötig Dämonen.
163:  ;********************************
164:
165:
166:  (defun berechne-leistung (rahmen schlitz)
167:     (let ((alter (rhol-w-s rahmen 'alter)))
168:               (cond (alter (list (rpack rahmen
                 'km-leistung 'wert (* 20000.0
                 (car alter))))))))
169:
170:  ;Beispiel eines Wenn-noetig-Dämonen für die
       Automobildatenbank.
171:  ;******************************************
172:
173:
174:  ;Datenbank über Automobile
175:
176:  ;Wissen über Lancia's

177:  (rpack 'lancia 'ist 'wert 'auto)
178:  (rpack 'lancia 'typ 'standard 'thema)
179:  (rpack 'lancia 'km-leistung 'wert '15000.0)
180:  (rpack 'lancia 'alter 'wert 1)
181:
182:  ;Wissen über Opel
183:  (rpack 'opel 'ist 'wert 'auto)
184:  (rpack 'opel 'typ 'standard 'rekord)
185:
186:  ;Wissen über VW
187:  (rpack 'vw 'ist 'wert 'auto)
188:  (rpack 'vw 'typ 'standard 'golf)
189:
190:  ;Wissen über Autos allgemein
191:  (rpack 'auto 'ist 'wert 'lancia)
192:  (rpack 'auto 'ist 'wert 'opel)
193:  (rpack 'auto 'ist 'wert 'vw);
194:  (rpack 'auto 'km-leistung 'wenn-noetig
       'berechne-leistung)
195:  (rpack 'auto 'alter 'wenn-noetig 'frage)
196:
197:  ;Datenbank zum Analogieschluß
198:  ;Fred ist wie ein Bär
199:
200:  ;Rahmen über Fred
201:  (rpack 'fred 'ist 'wert 'mann)
202:  (rpack 'fred 'wohnung 'wert "Hauptstr. 15")
203:  (rpack 'fred 'groesse 'wenn-noetig 'wie-baer)
204:  (rpack 'fred 'gangart 'wenn-noetig 'wie-baer)
205:
206:  ;Rahmen über Teddy Bär
207:  (rpack 'ted 'ist 'wert 'baer)
208:  (rpack 'ted 'wohnung 'wert 'hoehle)
209:  (rpack 'ted 'groesse 'wert 'gross)
210:  (rpack 'ted 'gangart 'wert 'polternd)
211:  (rpack 'ted 'nahrung 'wert 'honig)
212:
213:  (defun wie-baer (rahmen schlitz)
214:          (let ((wert (car (rhol 'ted schlitz '
               wert))))
215:                   (cond (wert (list (rpack rahm
                    en schlitz 'wert wert))))))
216:
217:
218:  ;Informationen über Computer
219:  (rpack 'atari 'modell 'wert '1040st)
220:  (rpack 'atari 'speicher 'standard '1mb)
221:  (rpack 'atari 'speicher 'wert '2mb)
222:  (rpack 'atari 'monitor 'wert 'color)
223:  (rpack 'atari 'monitor 'standard 'sw)
```

Literatur

[1] Finin, T. Implementing PFL. Part 1 & 2. AI Expert, November & December 1986. CL Publications, Palo Alto, CA.

[2] Lenat, D.B. Software für künstliche Intelligenz. Spektrum der Wissenschaft. Sonderheft Computersoftware 1985.

[3] Sarnow, K. Elemente der künstlichen Intelligenz. 1. Teil: Atome und Listen, ST-Computer, 3/87, p.37ff.

[4] Winston, P.H. & B.K.P. Horn. LISP. 2nd Edition. Addison Wesley Publishing Company. Reading, Massachusetts. 1984.

Dies war der letzte Teil der Serie 'Künstliche Intelligenz'. Falls Sie Interesse an dem Thema KI haben, dann schreiben Sie uns was Sie aus dem Bereich der KI interessieren würde. Wir versuchen darauf einzugehen und je nach Nachfrage weitere, auch praxisbezogene Artikel zu veröffentlichen. Natürlich sind auch Sie aufgefordert, eigene Beiträge zu liefern. Zuschriften bitte direkt an:

MERLIN-Computer GmbH
'KI'
Industriestr. 26
6236 Eschborn

Zertifikate selbstgemacht

Seit ein paar Jahren schon gibt es auch auf anderen Computern Programme wie PRINT SHOP oder PRINT MASTER, mit denen es möglich ist, Grußkarten, Schilder oder ähnliches zu kreieren. Nun ist ein Programm, das es schon länger für den IBM PC gibt, auf den ATARI ST umgesetzt worden. Inwiefern unterscheidet es sich von seinen Mitwerbern und was bietet es Neues gegenüber beispielsweise PRINTMASTER, der auf dem ST erhältlich ist ?!

CERTIFICATE MAKER, das von der Firma Springboard, die durch Programme wie ART A LA MAC auf dem Macintosh und NEWSROOM PRO auf dem IBM bekannt worden ist, vertrieben wird, gehört zu der gleichen Gattung von Programmen wie PRINTMASTER und PRINT SHOP. Mit CERTIFICATE MAKER lassen sich Urkunden und Zertifikate für (fast) jeden Zweck erstellen. Dies geschieht zunächst durch das Aussuchen einer gewünschten Urkunde aus einer Vielfalt von 220 vorgefertigten Ideen dafür (siehe Bild), die noch zusätzlich mit persönlichen 'warmen Worten' verfeinert werden können. Für diese Texteinträge bietet CERTIFICATE MAKER fünf verschiedene Zeichensätze an, die auch zwischen einer Größe von 16 und 32 Pixel einstellbar sind. Mehr Auswahl bietet das Programm aber in der Wahl der Umrandungen der Urkunden, denn es gibt dicke, dünne, viktorianische, persische Ränder und Ränder mit Bären oder Enten: ein Beispiel aus dem Reservoir von zwanzig Randarten finden Sie als Umrandung der Urkunden auf dieser Seite.



Bild: Einige Beispiele der mitgelieferten Urkunden

Sö könnten auch Ihre Ausdrucke aussehen!

Die Motive der Urkunden sind professionell und mit viel Liebe gezeichnet worden. Ursprünglich ist das Programm sicherlich für ernste Anwendungen zum Beispiel für eine Urkunde für den nettesten Angestellten oder eine Auszeichnung für den bestandenen Führerschein erdacht worden, allerdings liegen die Stärken eher auf den netten Anspielungen und witzigen Neckereien. Wie wäre es zum Beispiel mit einer Urkunde für den 'Mister Alleswisser' oder für 'die Person, mit dem größten Mund' oder wollen Sie den lustigsten Mitarbeiter als 'Company Clown' auszeichnen ? Viele Urkunden sind recht lustig und können oftmals als Anspielung auf irgendwelche Eigenheiten von Personen benutzt werden. Im Bereich SPORT sind besonders viel Urkunden vorhanden, so daß Sie ab sofort die Urkunden in ihrem Club selbst erstellen können – leider fehlen modernere Sportarten wie zum Beispiel Surfen. Eine mehrfache Ausfertigung von Urkunden läßt sich besonders einfach herstellen, da nach Erstellung einer Namensliste diese Namen an ganz bestimmte durch das Wort "name" in der Urkunde gekennzeichnete Stellen eingesetzt und ausgedruckt werden – eine gute Idee besonders für den Bereich von Wettkämpfen.

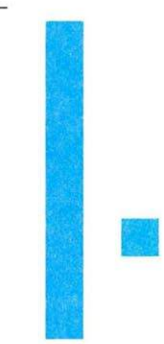
Der Wermutstropfen an diesem von der Idee und Grafik her sehr schönen Programm, ist seine nicht gerade übermäßige Schnelligkeit. Obwohl es schon um einiges besser als PRINTMASTER programmiert

worden ist, merkt man doch, daß es sich einmal mehr um eine Anpassung an den ST handelt. So sind auch eine große Anzahl von Druckertreibern vorhanden, aber vergeblich sucht man Druckertypen wie NEC P6, also 24-Nadel-Drukker, oder etwa STAR NL 10, der aber nach einigem Ausprobieren mit einer anderen Druckeranpassung 'zum Laufen' gebracht wurde. Leider wird die Urkunde nach jeder Eingabe neu aufgebaut, was doch eine kleinen Moment dauert und bei vielen Änderungen verärgern könnte – einige Routinen, in Maschinensprache geschrieben, könnten nicht stören. Leider gibt es auch keine Möglichkeit das Bild zeichnerisch nachzubearbeiten, geschweige denn eigene Ideen zu verwirklichen. Glücklicherweise gibt es eine – käufliche zu erwerbende – Zusatzdisk, auf der über hundert neue Urkunden und 24 neue Ränder zu finden sind; man darf hoffen, daß diese nicht zu teuer sein wird.

Nichtsdestotrotz macht das Programm beim Erstellen der Urkunden viel Spaß und die meisten der Urkundenempfänger freuen und amüsieren sich darüber, zum Beispiel die Person mit dem 'schönsten Lächeln' oder ein 'Party Tier' zu sein....

SH

Softline
Schwarzwaldstr. 8a
7602 Oberkirch
Tel.: 07802/3707
DM 98,–



Talkshow

...oder wie man eine Applikation zum Reden bringt

Viele Programmierer verspüren immer wieder den Wunsch, Daten zwischen verschiedenen Programmen auf 'legale' Weise austauschen zu können. Leider ist dieser Datenaustausch nicht genormt. Trotzdem ist es möglich, wenn die Programme aufeinander abgestimmt worden sind, Daten über GEM (unglaublich, aber wahr) auszutauschen. Dies soll unser Thema in diesem Monat sein.

Einige der im GEM vorhandenen Befehle sind inzwischen weitläufig bekannt und beschrieben worden, und trotzdem findet man immer wieder eine gewisse Scheu und Unsicherheit, wenn es um die Anwendung ganz bestimmter Befehle geht. Zu diesen Befehlsgruppen gehören die APPL--, SHELL-- und SCRAP--Befehle. Mit den SHELL-- und SCRAP--Befehlen werden wir uns ein anderes Mal beschäftigen. Diese ST-Ecke soll den Befehlen Appl-write(), Appl-read() und Appl-find() gewidmet sein. Zu Appl-tplay() und Appl-trecord() läßt sich folgendes sagen: Selbst nach längerem Ausprobieren (von einigen Programmierern) ist es uns nicht gelungen, diese Routinen vernünftig zum Laufen zu bekommen. Es scheint so, als würden nicht die Aktionen, sondern die Adressen der Routinen, die diese Aktionen ausführen, aufgezeichnet werden, was natürlich völlig unbrauchbar ist! Sollte es dennoch einem unter Ihnen gelungen sein, eine vernünftige Anwendung aus diesen Routinen herauszufinden, bin ich natürlich sehr daran interessiert.

Nun aber zu den guten Nachrichten: Die anderen Appl--Routinen funktionieren hervorragend. Zu **Appl-init()** gibt es eigentlich recht wenig zu sagen. Diese Routine meldet Ihr Programm bei GEM an und sorgt so dafür, daß Sie Routinen des AES ausführen können. Dabei revanchiert sich GEM mit einer Identifikationsnummer. An sich sollte dieser Wert als Rückgabeparameter im Register D0 zu finden sein, was aber nicht der Fall ist. Stattdessen existiert eine schon vordefinierte Variable gl-apid (**glo**bale **Ap**plikations-**Id**entifikation), die sie (unter C) nur noch als 'extern' deklarieren müssen, und in der dieser Wert zu finden ist. Sie werden im weiteren Verlauf des Artikel noch erkennen, welchen Zweck diese Nummer erfüllt. **Appl-exit()** hat den Sinn, die Applikation wieder abzumelden und die ID-Nummer wieder freizugeben. Beachten Sie aber, daß selbst dann, wenn Sie keine Verwendung für die ID haben, trotzdem vor dem Aufruf eine AES-Routine appl-init() durchgeführt werden muß, da sonst ihr GEM-Programm einfach abstürzt.

Nun wollen wir aber zu den interessanteren Routinen kommen, mit denen auch Informationen ausgetauscht werden können. Das Prinzip ist eigentlich ganz einfach: Weiß man die Applikations-Identifikation eines anderen Programmes - beispielsweise die einer Accessory, so kann man diesem mit appl-write() Daten über eine sogenannte Pipeline schicken. Durch die Angabe der ID sorgt GEM dafür, daß diese Informationen nur an diese spezielle Applikation übermittelt wird.

Wo ist sie denn...?

Natürlich werden Sie aber einwenden, daß Sie diese ID nicht wissen, denn die Programme teilen sie einem nicht mit. Das ist aber weiter kein Problem, sofern Sie wissen, unter welchem Programmnamen ihr anzusprechendes Programm geladen worden ist. Diesen Namen merkt sich GEM in Zusammenhang mit seiner ID. Um diese ID zu erfahren, benutzen wir den Befehl Appl-find(), dessen Parameter folgendermaßen aussehen:

ap-id =
appl-find(ap-filename);

int ap-id:
ID, der gesuchten Applikation
char* ap-filename:
Dateiname, der gesuchten Datei - auf 8 Zeichen erweitert (siehe Text)

Zu dem Dateinamen sind nun folgende Bemerkungen zu machen: Der Extender wie auch der Pfad des Dateinamens sind unwichtig und müssen daher entfernt werden. Sollte der übriggebliebene Name kürzer als 8 Buchstaben sein, muß er mit Spaces aufgefüllt werden. Weiterhin müssen die Buchstaben, falls es kleine sind, in große umgewandelt werden. Ist dann dieses Programm geladen worden, findet man sicher die entsprechende Applikations-Identifikation. Wie verschickt man nun Nachrichten an andere Applikationen ?

Dazu schauen wir uns noch einmal die Struktur des Ereignis-Puffers an, der bei evnt-mesag() oder auch evnt-multi() eine entscheidende Rolle spielt. Dieser Puffer ist ein Feld, das aus sechzehn Bytes besteht:

Wort 0:
Enthält das Kommando, an dem die Applikation erkennt, um welches Ereignis es sich handelt. Beispielsweise eine Redraw-Meldung oder eine Meldung, daß ein Menü angeklickt wurde.

Wort 1:
Dieses Wort enthält die ID des Prozesses, der die Meldung geschickt hat.

Wort 2:
Ist dieses Wort ungleich 0, liegen mehr als 16 Werte vor, die dann auch gelesen werden müssen. Mehr dazu im Text.

Appl-write() gibt uns nun die Möglichkeit, in die Pipeline hineinzuschreiben, die dann an die bestimmte Applikation weitergeleitet wird. Das Empfangen der Daten wird unten näher beschrieben. Wie eben erwähnt, handelt es sich dabei um eine sechzehn Byte lange Botschaft. Zunächst also schreiben wir in das erste Wort (zwei Bytes) die Nummer der Botschaft. Dabei ist zu beachten, daß diverse Botschaftsnummern von GEM schon belegt worden sind: Dies sind die Nummern 10, 20-29, 40 und 41. Die Bedeutung dieser Botschaften schauen sie bitte in der einschlägigen Literatur nach, da sie den Rahmen der ST-Ecke sprengen würde. Denkbar wären also alle anderen Zahlen, so könnte beispielsweise eine 50 bedeuten, daß Sie Koordinaten übertragen wollen, während eine 111 eine Übertragung eines Zeigers, der auf einen Text zeigt, ankündigt. In das zweite Wort schreibt man die ID des sendenden Programms, so daß der Empfänger eventuell herausbekommen kann, wo die Nachricht herkommt. Nun kommt es vielleicht vor, daß Sie einmal eine Nachricht senden wollen, die größer als 16 Bytes ist. Dies erkennt der Empfänger an Wort 2! Ist der Inhalt von Wort 2 gleich Null, bedeutet das, daß die Nachricht 16 Bytes lang ist; wollen Sie aber beispielsweise 35 Zeichen senden, so schreiben Sie in dieses Wort die Zahl 19 - also 35-16 Bytes. Dies würde bedeuten, daß der Empfänger außer den 16 Bytes noch zusätzliche 19 Bytes lesen muß.

Achtung Umleitung!

Der Nachteil ist, daß diese ganzen Bytes über den Umweg GEM erst an ihren Empfänger gelangten. Eleganter ist folgende Idee. Sie kopieren ihre zu übertragenden Daten in ihrem Sendeprogramm zusammen in einen Puffer. Dann übertragen Sie nur die Adresse ihres Puffers. Dadurch werden über den Umweg GEM nur wenige Bytes -die Adresse- übertragen. Da aber der Sender die Adresse der Daten, die im Speicher zu finden sind, kennt, kann es sich diese direkt aus demselben holen. Anhand des Sende-Beispielprogramms können Sie erkennen, daß ab dem vierten Feldelement die Adresse des Textes nicht aber der gesamte Text eingeschrieben und dann übertragen wird.

Nachdem Sie wissen, wie das Feld des Puffers zu füllen ist, wollen wir uns dem Senden, das sehr einfach ist, zuwenden. Dazu verwenden wir **Appl-write()**, das wie folgt definiert ist:

ap-ret = appl-write(ap-id-empf, laenge, puffer)

Das Neuste aus unserer Sammlung

Am Ende dieses Jahres möchten wir Ihnen zum letzten Mal in diesem Jahr die neusten Disketten unserer Sammlung vorstellen. Auch diesen Monat sind wieder einige Lekkerbissen dabei.

Wir möchten uns bei all den vielen Programmautoren bedanken, die mit viel Mühe ihren Beitrag zu der Sammlung lieferten. Wir sind stolz darauf, daß in unserer Sammlung keine Software aus anderen Sammlungen, aus USA oder sonstwo enthalten ist, sondern nur eigene, die uns unsere Leser zuschickten.

Auch wir haben uns bemüht, die Disketten so schnell wie möglich zu liefern. Das funktionierte auch grundsätzlich innerhalb einer Woche. Doch machte uns unser Diskettenlieferant FUJI einen Strich durch diese Rechnung, da er mehrere Wochen lang nicht liefern konnte. Aus jenem Grunde stiegen wir auch auf andere, nicht minder gute, Markendisketten um.

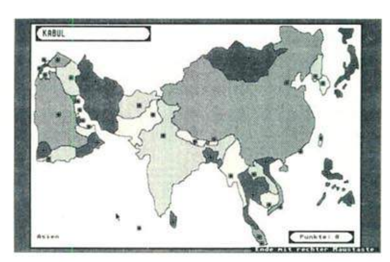
Wir hoffen, daß Sie mit dem Service bisher zufrieden waren und wünschen Ihnen ein gutes neues Jahr.

Ihre ST-Computer Redaktion



110
Utilities
● *R-Copy:* Automatischen Backup-Programm von Ramdisk auf Diskette. Vorhandene Dateien werden dabei umbenannt. Kopieren in beliebige Ordner. Incl. Source in Pascal. ● *Pashelp:* Crossfeferenzprogramm für Pascal. Schlüsselwörter gross. Ausgabe normal oder als Tempusfile. ● *Unidruck:* Druckprogramm unter Berücksichtigung der Sonderzeichen. Mehrere Drucker. Zeilenummerierung. Incl. Source in Pascal. ● *Font:* Programm zum Einbinden von DEGAS-Fonts in PascalProgramme. Incl. Source in Pascal.

111
Utilities
● *Basicref:* Crossreferenzprogramm für GFA-Basic. Hilft mit vielen Optionen Fehler aufzudecken. Incl. Source in GFA-Basic. (s/w) ● *MSK-Edit:* Maskeneditor für GFA-Basic. Generiert mit einfacher Bedienung eine komplette Eingabemaske als Quelltext (Form Input). Incl. Source in GFABasic. (s/w) ● *Retten:* Speichert zu lange Dateien in mehreren Teilen auf verschiedene Disketten. Incl. Source in Fortran. ● *SECO:* Verwaltet Bildausschnitte (mit GFA-Basic Befehlen PUT GET bearbeitbar) und fügt sie zu Bibliotheken zusammen. Incl. Source in GFA-Basic.





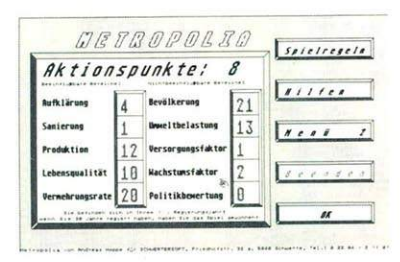
112
Erdkunde
● *WORLD:* Lernprogramm. Fragt alle Länder und Hauptstätte der ganzen Welt ab. Dabei erfolgt die Antwort per Mausklick auf das grafisch dargestellte Land. Enthaltene Landkarten: BRD, USA, Mittelamerika, Südamerika, Europa, Asien, Afrika und Ozeanien.

113
● *Zeitmanager:* Semiprofessionelle Terminplanverwaltung. Eigener Desktop, Bedienung durch Icons (ähnlich DISKSTAR), erinnert an Termine, druckt Listen, verwaltet komplettes Timing, automatische Verwaltung von fixen Terminen, eingebauter Kalender, sehr flexibel und leicht zu bedienen. (s/w)



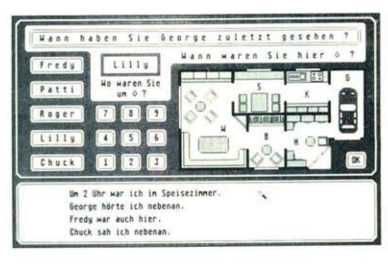


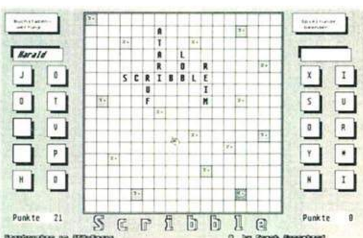
114
Spiele
● *Metropol:* Wirtschaftsspiel. Durch Manipulation bestimmter Faktoren haben Sie Einfluß auf die Wirtschaftslage, die sich im Versorgungsfaktor, Umweltbelastung, Versorgungs- und Wachstumsfaktor, sowie letztlich in der Politikbewertung ausdrückt. Neigt sich ein Punkt ins Extreme, so kann es Ärger geben und die Wirtschaftslage kriseln. Das zu verhindern ist Ihr Job. (s/w) ● *Dallas:* Hierbei dreht es sich um Öl (worum auch sonst). Sie als JR kämpfen um Profit und gegen Ihre Konkurrenten. (s/w)





115
Spiele
● *Sherlock:* Die Aufgabe ist es, einen Mord aufzuklären. Dazu müssen Sie als Sherlock Holmes die beteiligten Personen befragen. Dies geschieht per Mausklick auf die Personen, Räume und Uhrzeit. Doch wehe Sie klären den Fall nicht auf, der Spott ist mir Ihnen. ● *El Bozo City:* El Bozos City of Bounds ist ein Textadventure mit interessantem Parser. Geübte Abenteurer dürften keine Schwierigkeiten haben, doch steckten viele Gags in der Story. ● *Scribble:* Eine schöne Scrabble-Version für den ST. Ein oder zwei Personen versuchen sich daran, möglichst viele Wörter auf einzelnen zufälligen Buchstaben zu bauen. (s/w) ● *Kreuzwort:* Programm zum Generieren eines Kreuzworträtsels. (s/w) (s/w) – Nur Monochrom (f) – Nur Farbe





Updates

Die folgenden Programme wurden aktualisiert. Ab sofort sind auf den Originaldisketten die neuen Versionen enthalten. Damit Sie nicht mehrere Disketten neu bestellen müssen, haben wir die Updates auf den Disketten U3 und U4 zusammengefasst.

U3

Autocopy (74)
Datebook (74)
Profilramdisk (77)
Konvert (58)
Statistik (76)

U4

Label Expert (79)
Vokabeltrainer(22)
Grafik (29)
Quiz (91)
Video (16)

Weiterhin wurde Diskette 49 (Volksforth 3.80), sowie Diskette 48 (Word 2.1, Spelling.Dic für Wordplus) aktualisiert.

Die Updatedisketten sind zu den gleichen Bedingungen wie die normalen PD-Disketten lieferbar (pro Stück DM 10,-).



Die älteren Disketten sind auch weiterhin erhältlich. Schauen Sie dazu bitte in einer der vorherigen Ausgaben nach oder fordern Sie die Liste an (frank. Rückumschlag).

Diskette 60
● *Synthi:* Erzeugt div. Sounds, die abgespeichert und über 'DOSOUND' in eigene Programme übernommen werden können (inc. der Source-Code) (s/w). ● *Waschi:* Waschhilfsprogramm für Junggesellen (s/w). ● *Expert:* Ein einfaches, lernbegieriges Expertensystem. ● *Adressen:* Adressverwaltungsprogramm mit Etikettendruck und einfacher Textverarbeitung.

Diskette 61
● *Hacomini:* Erzeugt Hardcopies in Miniaturformat (48mm * 68mm) (s/w). ● *Helper:* Hilfsprogramm bestehend aus Diskmonitor und Druckerutility. ● *Bidipo:* Scrollt eine Bilderkette über den Schirm. Ideal für eigene Diashows oder Werbegrafiken. ● *Datobert:* Grafische Darstellung. Linien- und Kuchengrafik, sowie in dreidimensionaler Balken- und Flächengrafik. Überlagerung einzelner Datenserien (s/w). ● *Direct all:* Directorydruck. ● *Überweisung:* Überweisungsdruckprogramm aus ST 6/87.

Diskette 62
MIDI Diskette
● *MIDIDRUM:* Rythmusmaschine mit leicht und vielseitig editierbaren Sounds. ● *MIDIPLAY:* Erlaubt das Einspielen, Speichern und Ausgeben von Midi-Sequenzen, incl. Source-Code in GFA-Basic.

Diskette 63
Spielesammlung
● *Fuffy:* Sehr schön gestaltetes Kniffel-Spiel mit zusätzlicher Auswertegrafik (s/w). ● *Snake:* Eine Schlange schlängelt sich durchs Leben, auf der Suche nach Fressen. Man achte auf Hindernisse (s/w). ● *Bomber:* Scramble-Spiel (f). ● *Reversi:* Der Name sagt alles (f).

Diskette 64
● *E-PLAN:* Konstruktionsprogramm für elektronische Schaltungen. Alle elektronischen Bauelemente sind vorhanden. Schaltung können auch beschriften und die Bauteile in die Stückliste übernommen werden. (s/w)

Diskette 65
Mal- und Zeichenprogramme
● *Columbia:* Malprogramm mit vielen Extras. Mehrere Bildschirme, Blockoperationen, Spiegeln, Verzerren, Drehen, Lupe. (s/w) ● *Paintlux:* Malprogramm mit zusätzlicher Bibliotheksverwaltung. Dadurch können z.B. elektronische Bauteile aus der Bibliothek in das Bild eingefügt werden. (s/w)

Diskette 66
Spieledisk
incl. Source-Code in OMIKRON Basic
● *MAZIACS:* Abenteuerspiel. Unser goldsuchender Held in Labyrinth, im Kampf mit bösartigen Spinnen und anderen Hindernissen (s/w). ● *Quadrat:* Bilderpuzzle. Ordnung im Caos (s/w).

Diskette 67
Utilities
● *Diskchecker* Prüft eine Diskette auf Fehler. Formatierund Schreib-/Leseprüfung (s/w). ● *DEGAS ELITE Fontmaker:* Erzeugt das DEGAS ELITE File Assign.SYS. Damit wird das Einbinen von neuen Fonts stark vereinfacht. ● *Printset* Druckeranpassungshilfe. Sonderzeichenausgabe. ● *TESTBILD* Das Farbmonitortestbild. ● *ICONEDIT* Sprite-, Maus- und Füllmustern. Übernahme in eigene Programme.

Diskette 68
Mal- bzw. Zeichenprogramme
● *Studio* Zeichnen in DIN A4-Format. Blockverwaltung. Automatische Bemaßung. Ausschnittshardcopy. ● *Van Gogh* Mal- und Animationsprogramm. 64 bewegte Blöcke. Incl. Blockeditor (f)

Diskette 69
● *Astrolabium* Zeichnen von Sternkarten- oder Himmelsausschnitts – zu beliebiger Zeit, an beliebigem Ort. Einstellbare Blickrichtung, – Höhe, und -winkel, Incl. Daten von ca. 9000 Sternen (s/w)

Diskette 70
● *Orbit* Zeigt die Laufbahn von Satelliten als bewegte Grafik über der Weltkarte. Z.B. für Funkamateure (s/w). ● *High End* Programm zum Berechnen von Lautsprecherboxen aller Art. Lautsprecher und Boxenmaße. Datei für Speaker enthalten. (s/w)

Diskette 71
Spieledisk
● *Ballerburg* Zwei verfeindete Burgherren zerstören sich gegenseitig Ihre Burgen. Steuerung der Kanonenwinkel und der Pulvermenge. (s/w) ● *Stagger* Reversiumsetzung. (s/w) ● *Hotelier* Hotelverwaltungsspiel. Leiten Sie Ihr Hotel und kämpfen Sie gegen die Konkurrenz. (s/w) ● *Roulette* Was mag dies wohl sein? (s/w)

Diskette 72
● *ZREF* Crossreferenz für C-Programme. Erzeugt Variablenliste mit Funktionsaufrufen, Typen, defines, structs, ..., Zeilennummern und vielen anderen Optionen. ● *Plotter* Werteplotter für wissenschaftliche Anwendungen. Lineare-, Lagrange-, 3-Punkte Interpolation sowie Regressionsanalyse. Nachträgliches Editieren (s/w). ● *Nicelist* Ausdruck von Source-Codes mit Seitenummern, Dateiname und Datum. Sonderzeichenanpassung.

Diskette 73
● *Videoarchiv* Videokasettenverwaltung. Leicht bedienbar. Mit Etikettendruck. (s/w) ● *Lohnsteuer* Lohnsteuerberechnungsprogramm ● *Heimmanager* Kontrolle über die private Haushaltsabrechnung. Jahresübersicht. (s/w)

Diskette 74
Ramdiskpaket u.a.
● *MAXIDISK* Die Ramdisk. Resetfest (Inhalt nach Reset noch vorhanden). Komprimiert seinen Inhalt (Eine 500K-Ramdisk kann wesentlich mehr aufnehmen als 500K). Frei wählbare Größe. Frei wählbare Laufwerkskennung. ● *COPY MAXIDISK-Utility.* Kopiert Inhalt eines Ordners beim Systemstart in die Ramdisk. ● *TIME MAXI-DISK-Utility* zum Stellen des Datums und Uhrzeit. ● *Mono.Neo* Speichert bei Alternate+Help den Bildschirm auf Diskette. ● *Autocopy* Ähnlich dem Programm COPY. Kopiert bestimmte Dateien in beliebige Ordner Datebook Erinnert beim Systemstart an wichtige Termine oder Geburtstage. u.a.

Diskette 75
Fontdiskette zu Fontmaster (PD40)
Neue, bzw. erweiterte Fonts, und Rahmen. Ergänzung zu Fontmaster (PD40).

Diskette 76
Schulungssoftware
● *Periode* Periodensystem der Elemente. Chemielernprogramm mit vielen Optionen. Mit allen chemischen Elementen. (s/w) ● *Statistik* Grafische Darstellung und Datenauswertung. (s/w)

Diskette 77
Utilitydisk
● *FCOPY 2.0* Schnelles Kopierprogramm. Formatieren, Verify, Multiple Copy, Scan und Teilkopie. ● *Speeder* Verdoppelt Schreib- und Lesegeschwindigkeit. ● *Filecopy* Kopieren von Einzeldateien. Auch mit einer Floppy. ● *Formater* Erhöht die Diskkapazität ● *Fastformat* Formatiert Schnelladedisks. Durch besondere Anordnung der Sektoren, wird immer schneller geladen. ● *Taskcopy* Diskbackup im Multitasking ● *Profilramdisk* Resetfeste Ramdisk mit Ausführungsprofil

Diskette 78
Utilitydisk
● *AESLIB* Bibliothek für GFA-Basic mit Autolinker ● *LINE-A* Line-A Binding für C-Programme. ● *Conv.WP* Konvertiert ASCII-Dateien in WP-Code von 1st Word. ● *Musik* Ähnliches Programm aber als Accessoir. ● *Setz-Uhr* Programm zum Stellen der Zeit+Datum incl. Source in Assembler.

Diskette 79
Etikettendruck
● *Label Expert* Druckt alle Art von Etiketten. 3,5 Zoll- 5,25 Zoll-Disketten, Audio-Kasetten, Video-Kasetten, Ordner, u.a. (s/w) ● *Grafikett* Bedruckt Disklabels auch mit Grafik, welche aus beliebigem Grafikbild ausgeschnitten wird. (s/w) ● *Etikettdat* Druckt Adressenaufkleber. (s/w) ● *Etikette.GFA* Luxuriöses Disklabeldrucker. Mit vordefinierter, artbezogener Grafikeinbindung. (s/w)

Diskette 80
Spieledisk
● *U-Boot* U-Boote orten (s/w) ● *Arger* 'Mensch ärger dich nicht'. (s/w)

Diskette 81
Malprogramm
● *Public Painter* Malprogramm mit interessanten Fähigkeiten, gepacktes Format, SNAPSHOT-Bilder, Lassofunktion, Blockmanipulation, Vergrößern, Verkleinern, Drehen, Biegen, Verzerren, Überschneiden. Drucken von Bildausschnitten. (s/w)

Diskette 82
Malprogramm
● *Masterpainter* Läuft in allen drei Auflösungen. Konvertiert jede Auflösung in beliebige andere. Blockbearbeitung. Vergrößern, Verkleinern, Rotieren, etc., bis zu 24 Screens. Ausschnittdruck. ● *Little Painter* Malprogramm . Rasteroperation, Musterspray, Verwischen, Drehen, Biegen, Blockladen und – speichern, Fontlader. (s/w)

Diskette 83
● *Starofix* Zeichsatzeditor für Down-Load Zeichen. Star NL10 und kompatible. Zeichensatz per Autostartprogramm in Drucker laden. Viele Fonts. ● *Starlist* Ausdruck von ASCII-Dateien mit Sonderzeichenwandlung.

Diskette 84
● *Druckedit* Zeicheneditoren für 1st Word Druckeranpassung ● *Draft Pin* Zeichen werden editiert und automatisch in beliebiges .HEX-File eingebaut. (s/w)

Diskette 85
Spieledisk
● *Skat* Eine Runde SKAT gegen zwei von ST gesteuerten Zocker. (s/w) ● *Delta* Gehirnmuskeltraining. Ähnlich dem Zauberwürfel. (f) ● *Grusel* Grafik- und Sounddemo. Geisterstunde auf dem Friedhof. Skelette, Abte und Gespenster treiben ihr Unwesen. (f)

Diskette 86
● *Carpet* Berechnungsprogramm von 3D-Funktionen. Hidden-Line Algorithmus und Beleuchtungseffekten. Funktionseingabe und frei einstellbare Parameter. (s/w) ● *3D-Plot* Ähnliches Programm in compiliertem GFA-Basic, mit einigen Extra-Funktionen. (s/w) ● *Funktionsplot* Funktionseingabe im Programm, verschiebbare Achsen, mehrere Darstellungsarten (s/w)

Diskette 87
● *Sternbild* Zeigt den Sternenhimmel, oder bestimmte Ausschnitte zu verschiedenen Zeiten. Planetennamen auf Anklicken. (s/w und Farbe). Räumliches Rot-Grünbild im Farbbetrieb.

Diskette 88
Druckertreiber
● *NEC-EMU* Der bereits in den NEWS angekündigte Hardcopytreiber für NEC P5/P6/P7. Erzeugt Hardcopies in 12 verschiedenen Größen und versch. Kontrasten. Enthält Druckerspooler. ● *LQ 800* Hardcopytreiber für EPSON 24 Nadeldrucker in drei verschiedene Größen. ● *NEC-Code* Ein Programm zur Einstellung des NEC P6 /P7 u.ä.. ● *SIMPLE* Komfortables Hardcopy-Programm mit vielen Extras. ● *STX-80* Hardcopytreiber für diesen preiswerten Thermodrucker. ● *WORDPLUS* Druckertreibersammlung für folgende Drucker: NL10, FX1000, SP1000I. ● *1st prop:* 1st.Proportional Treiber für folgende Drucker: HR15, FX85.

Diskette 89
● *Führerschein* Führerscheinrtraining. 500 Fragen zur theoretischen Prüfung. ● *Bundesliga* Fußballmanager. Errechnen von Tabellen, editieren aller Spiele von der 1., 2. Bundesliga, bis hin zur untersten Bezirksliga.

Diskette 90
Utilities
● *TOPS* The Other Pascal Shell (für PascalPlus), TEMPUSfähig, erzeugt Crossreferenz. Kopieren, Löschen und Umbenennen von Files. ● *SIGNUMShell* Eine Shell zu Signum. Umschaltung zwischen Signum Editor, den verschiedenen Druckprogrammen, den Zeicheneditoren per Menüleiste. Automatisches Install. ● *Tempelmon 1.8* Neue Version des weitverbreiteten Monitors. Läuft nun auch in Farbe. ● *Maxidisk 4Mb* Die komprimierende Maxidisk mit max. 4MByte Kapazität. Läuft auch auf Blittertos. Komprimiert automatisch ihren Inhalt. Resetfest. ● *RAMDISK* Gleiches Programm wie MAXIDISK, nur ohne Komprimieralgorithmus. ● *COPY* Beide Ramdisk enthalten ein Autocopyprogramm zum kopieren bestimmter Files in Ramdisk. ● *RCS.Wandel* Wandelt *.h Files nach Basic und Assembler.

Diskette 91
Lernprogramme
● *Pauk* Lerndatei mit komfortabler Steuerung, eigener Dateneingabe, wiederholung falscher Fragen nach Lernkonzept. (s/w) ● *Hypervoc* Sehr schön gestaltetes Vokabellernprogramm. (s/w) ● *ST-Dic* Fremdsprachenlernprogramm mit großer Lateinbibliothek. (s/w) ● *QUIZ* Eine Art Trivial Persuite. Enthält Fragen mit drei möglichen aber nur einer richtigen Antwort. Eigene Fragen einbinden möglich. Enthält (ATARI ST-Einsteigerkurs). (s/w)

Diskette 92
Utilitydisk
● *Superbatch* Batch-Prozessor: bedingte Batchbearbeitung (if) Batch Protokoll auf Bildschirm, Drucker, Datei Verzweigung, Fehlerbehandlung. ● *ST-Klick* Multifunktionsaccessoir: Notizen, Wecker, Drucker Init, Schreibmaschine, Kalender, Taschenrechner (s/w) ● *XREF* Crossreferenz zu C ● *Term* Terminalprogramm mit Xmodem, Vollduplex, u.v.a.m. ● *Alfaformat* flexibles Formatierprogramm (9-11 Sektoren) (bis 86 Tracks).

Diskette 93
● *ADR-2* Semiprofessionelles Adressverwaltungsprogramm mit Serienbrieferstellung (mit und ohne 1st Word). Editierbare Druckeranpassung (Steuerzeichen, Zeichenwandlung), erzeugt Adress-Telefon und Geburtstagslisten. Druckt Labels, Etiketten nach freidefinierbarem Format. (s/w) ● *Dat-Text* Adressverwaltung für Mehrpersonenzugriff auf eine Datei. Datenzugriff nur durch Passwort (s/w)

Diskette 94
Spieledisk
● *MacPan* Eine schnelle Pacman-Version für den ST. Per Joystick ins Labyrinth auf der Suche nach Powerpills. Mit Empfehlung der Redaktion. ● *Solitar* Dreidimensionales Solitär mit sehr ansprechender Grafik. ● *Galaxy Spielmeister* Brettspiele zur Förderung der Gehirnaktivität (s/w)

Diskette 95
Spieledisk
● *Diamond Miner* Eine Spiel ähnlich Boulder Dash. Sehr schnelle Grafik und 30 Level. Eingebauter Feldeditor (s/w) ● *Snake* Zwei Schlangen versuchen sich den Weg abzuschneiden. (s/w) ● *Minenfeld* Ein wagemutiger Schatzsucher begibt sich durch verschiedene Minenfelder und muß versuchen die Minen zu umgehen. (s/w) ● *Missle* Missleattack (s/w)

Diskette 96
Spielesammlung
● *Andurlin* Geschicklichkeitsspiel mit 75 Bildern. (s/w) ● *Gilgalad* Arcade Adventure mit 224 Räumen. (s/w) ● *Invadors* Der Spieleklassiker. Außerirdische Raumschiffe greifen die Erde an. (s/w) ● *Minigolf* Minigolf auf dem ST. 18 Bahnen, Steuerung (Richtung, Anschlagstärke) per Maus. (s/w)

Diskette 97
● *Trash* The First Trash-Groove Adventure. Textadventure mit deutschen Befehlen. Parodie auf die Musikund Punkszene. (incl. Source in Omikron.-Basic) ● *Labyrinth* Im Labyrinth auf der Suche nach dem Ausgang. Sehr schöne 3D Darstellung. 2D Darstellung als Hilfe. Schnell und ruckfrei. (s/w)

Diskette 98
VAX GAMES
Spielesammlung
● *SNAKE* Eine Schlange auf Nahrungssuche. ● *QIX* Sie müssen 75% des Bildschirms einzäunen ohne gebissen zu werden. (Ähnlich STIX) ● *Nibbler* Eine Schlange schlängelt sich durch ein Labyrinth. ● *Pacman* Kommentar überflüssig ● *Doors* Pacman Variante mit Hindernissen ● *Robot* Vosicht Roboter

Diskette 99
● *CNC Simulation* Simulation einer CNC gesteuerten Fräsmaschine. Programmierbar. Grafische Anzeige (s/w)

Diskette 100
● *ST-Speech* Sprachausgabe auf Softwarebasis. Wandelt englischsprachige Texte automatisch in verständliche Lautschrift. (s/w) ● *Fortran Shell* Eine Shell für Fortran 77. Bequemes Aufrufen von Compiler, Linker, und anderen Programmen. ● *Life is Life* Life-Simultionsprogramm. Sechs verschiedene Feldgrößen, Drehen, Verschieben, Kopieren von Blöcken wie im Malprogramm, Festlegen der Fortpflanzungsregeln, hohe Geschwindigkeit. (s/w) ● *Fastlife* Life-Simulation in Volksforth. Wahnsinige Geschwindigkeit. (s/w) ● *GEMFRAC* Berechnung fraktaler Landschaften in Farbe und s/w. ● *Sharp* Basic-Programme für Sharp-Basicrechner. Geeignet für Direktüberspielung (mit Interface).

Diskette 101
Wissenschaftliche Anwendungen (Schwerpunkt Chemie Medizin)
● *Laokoon* Sumulation von NMR (Magnetische-Kern-Resonanz-Spektoskopie). Bestimmung von chemischen Verbindungen und Kopplungsarten. (s/w) ● *IR* Infrarot-Spektroskopie. Programm zur Strukturanalyse. (s/w) ● *Plotter 1.7* Grafische Darstellung von Werten. Lagrangelnterpolation, Spline oder Regressionsgrade. (s/w) ● *Plottit* Kurvenplotter. Direkte Funktionseingabe mit sofortiger Syntaxprüfung. Nullstellenberechnung. (s/w) ● *Ganglion* Programm zur Simulation eines neuralen Netzes (s/w)

Diskette 102
Programmierutilities
● *Fileselect* Eigene flexiblere Fileselectroutine. Zum Einbinden in eigene Programme. ● *Printf* Verbesserte PRINTF-Routine für Megamax C. ● *RCS-ICN* Generiert ein ICON aus einem Bildschirmbereich

(s/w) – nur Monochrom
(f) – nur Farbe

Sonderdisks

Die folgenden Programme sind nicht Public-Domain. Sie können aber bei uns bezogen werden.

A.) – *TOS:* Die letzte Disketten-Version des TOS vom 6. 2. 1986. Zum fehlerfreien Betrieb des Rechners benötigen Sie diese Version, da z. B. mit älteren Versionen manche Ordner nicht gelesen werden (0 Bytes free).
Unkostenbeitrag 15,– DM incl. Diskette.

B.) – *RCS:* Das Resource Construction Set aus dem ATARI-Entwicklungspaket. Unverzichtbar bei der Erzeugung von Objektbäumen, Dialogboxen, Drop-Down-Menüs. Urzum Ein Muß für GEM-Programmierung.
Unkostenbeitrag 15,– DM incl. Diskette.

Bis auf die Höhe des Unkostenbeitrags gelten die gleichen Versandbedingungen wie bei der Public-Domain-Software.

Versandbedingungen:

Sämtliche Disketten können Sie ab dem Erstverkaufstag direkt von der Redaktion erhalten. Wir haben für Sie den schnellstmöglichen Versandservice eingerichtet. Lieferung innerhalb einer Woche.

1. Anruf genügt:

ST-Computer Redaktion
Tel.: 06196/48 18 11

2. Schriftliche Bestellung:

– Der Unkostenbeitrag für eine Diskette beträgt DM 10,–
Bezahlung per Scheck oder Nachnahme nur im Inland).
– Bei Nachnahme zuzüglich DM 3,50 Nachnahmegebühr
– Fügen Sie dem Betrag folgende Versandkosten (Porto + Verpackung) bei: Inland DM 5,– Ausland DM 10,–
– Legen Sie bitte, falls zur Hand, einen Aufkleber mit Ihrer Adresse bei.

Die Bezugsadresse lautet:

MERLIN-Computer GmbH
ST-Computer-Redaktion
'PD-Service'
Postfach 59 69
D-6236 Eschborn

Bei Fragen bezüglich der Disketten wenden Sie sich bitte an die Redaktion.

int ap-ret:
Ist im Fehlerfall 0, sonst gibt das AES einen positiven Wert zurück.

int ap-id-empf:
Applikations-ID des Programmes, das die Daten empfangen soll. Diese ID haben Sie durch appl-find() ermittelt.

int laenge:
Dort steht die Anzahl der zu übertragenden Bytes, allerdings wird hier die volle Anzahl angegeben. Im Normalfall steht hier aber die Zahl 16.

char *puffer:
Hier setzen Sie die Adresse ihrer zu sendenden Daten ein.

Achten Sie bitte darauf, daß Sie als Länge keinesfalls eine Null einsetzen, sonst stürzt der Rechner ab. Jetzt sind die Daten abgeschickt, allerdings soll eine zweite Applikation diese Daten auch empfangen. Dies geschieht genauso, wie Sie auch sonst Botschaften empfangen, und zwar mit evnt-mesag() bzw. evnt-multi(). In diesen beiden Routinen benötigen Sie einen 16 Byte großen Puffer, und genau jener Puffer enthält ihre Botschaft. Rufen Sie beispielsweise evnt-mesag() auf, wartet dieses solange, bis es eine Botschaft empfängt. Dabei ist es völlig unerheblich, ob dies eine Botschaft ist, die vom Desktop stammt oder von einer fremden Applikation. Wichtig ist nur, daß etwas in die Pipeline geschrieben wurde, und auch genau diese Applikation gemeint war (siehe oben bei der Erklärung von appl-write()). Die Empfangs-Applikation muß eigentlich nur noch das erste Byte des Puffers überprüfen und schon weiß sie, welche Meldung sie bekommen hat und kann Sie entsprechend auswerten. Einen wichtigen Punkt darf man allerdings nicht außer acht lassen. Wurde eine Meldung geschickt, die länger als sechzehn Bytes ist, was man aber sehr elegant umgehen kann, so müssen diese restlichen Bytes auf jeden Fall gelesen werden, was mit **appl-read()** geschieht. Nach Erkennen, daß Wort 2 ungleich Null ist, führt man folgenden Befehl aus:

ap-ret = appl-read(ap-id, laenge, puffer)

int ap-ret:
ist im Fehlerfall 0, sonst ein positiver Wert.

int ap-id:
Die Identifikation der Applikation, die gelesen werden soll. Im Normalfall wird das die eigene Applikations-ID, also gl-apid sein, da man zusätzliche Daten aus der eigenen Pipeline liest.

int laenge:
Anzahl der aus der Pipeline zu lesenden Bytes.

char* puffer:
Adresse, an die die gelesenen Daten geschrieben werden sollen.

Der Anwendung dieses Verfahrens sind kaum Grenzen gesetzt. Es soll allerdings nicht verschwiegen werden, daß dieses Verfahren auch Nachteile hat: Daten können nur dann erfragt oder empfangen werden, wenn Sender und Empfänger aufeinander abgestimmt sind. So ist die einzige Normung durch die Befehle des GEM gegeben. Diese Befehle sollten daher von allen Applikationen verstanden werden. Denkbar wäre es auch, über diese Verfahrensweise einer Accessory vorzutäuschen, sie wäre angeklickt worden (AC-OPEN). Sie finden sicherlich einige Dinge, die ihren Reiz besitzen. Eine interessante Tatsache ist übrigens, daß bei DEGAS durch Angabe eines bestimmten Befehls durch eine Accessory über die Pipeline, die Adressen der im Computer befindlichen Bilder zu erfahren sind. Umso erstaunlicher ist es, daß das auch schon bei alten Versionen von DEGAS funktioniert. Denkbar sind sogar diverse Accessories, die miteinander kommunizieren.

Ich wünsche noch viel Spaß beim Austauschen von Daten und möchte nur noch kurz erwähnen, daß das häufig erfragte Problem, die Maussteuerung zu manipulieren, gelöst ist. Sie können sich daher auf dieses Thema freuen, das ich in den nächsten Monaten vorstellen werde.

SH

Listing 1: Das Sende-Programm

```
 1:
 2:  /**********************************************************/
 3:  /* Programmbeispiel zum Senden von Informationen von      */
 4:  /* Applikation zu Applikation.                            */
 5:  /**********************************************************/
 6:
 7:  #include <gemdefs.h>
 8:  #include <osbind.h>
 9:
10:
11:  extern long strlen();
12:
13:  extern int gl_apid;       /* ist schon vordefiniert */
14:  int id;                   /* Identifikationsnummer ID */
15:
16:  int mes1[]={42,0,0,1,2,3,4,5}; /* Botschaft an die */
                                    /* Applikation */
17:  char text[]="Dies ist eine längere Nachricht\n";
18:  long *poi;                /* Zeiger auf Feldelement */
19:
20:  main()
21:  {
22:     appl_init();           /* Applikation initialisieren */
23:
24:     id=appl_find("APP1    "); /* Suche 'APP1.ACC' im */
                                 /* Speicher und ermittle */
25:                              /* Application-ID */
26:
27:
28:  /* gebe eigene ID und die der Accessory aus */
29:     printf("meine id: %d    seine id: %d\n",gl_apid,id);
30:
31:
32:     if (id==-1) /* Applikation dieses Namens gibt es nicht */
33:     {           /* im Speicher */
34:        appl_exit(); /* GEM freigeben */
35:        Pterm0();    /* Programmende */
36:     }
37:
38:     mes1[1]=gl_apid; /* Hallo, wir sind's */
39:     mes1[2]=0;       /* weniger als 16 Bytes */
40:     appl_write(id,16,mes1); /* Ab geht die Post */
41:
42:  /* Zweite Nachricht vorbereiten */
43:     mes1[0]= 111;      /* eigene Botschaft, mit 111 */
                           /* gekennzeichnet */
44:     mes1[1]= gl_apid;  /* Absender nicht vergessen.... */
45:     mes1[2]= 0;        /* mehr als 16 Bytes sind's nicht */
46:
47:     poi=(long*)&mes1[3]; /* Zeiger auf Feldelement setzen */
48:     *poi=(long)text;   /* Adresse des Strings ist in mes[3] */
                           /* und mes[4] */
49:
50:     printf("PROGRAMM: %s Adresse: %lx\n",*poi,*poi);
51:     appl_write(id,16,mes1); /* Botschaft verschicken */
52:
53:     evnt_timer(1000,0);      /* Lassen wir der Accessory ein */
                                 /* wenig Zeit */
54:     appl_exit();             /* Wir melden uns ab */
55:
56:  }
```

Listing 2: Die Empfangs-Accessory

```
 1:
 2:  /**********************************************************/
 3:  /* Accessory-Beispiel zum Empfang von Informationen von   */
 4:  /* Applikation zu Applikation.                            */
 5:  /**********************************************************/
 6:
 7:  #include <osbind.h>
 8:  #include <gemdefs.h>
 9:
10:  extern gl_apid;           /* extern deklariert */
11:  int msg[16];              /* Ereignis-Puffer */
12:  long *poi;                /* Zeiger auf Feldelement */
13:  char name[80];            /* Hilfsstring */
14:
15:  main()
16:  {
17:
18:     appl_init();           /* gl_apid enthält die Nummer, */
                               /* nicht D0 */
19:     menu_register(gl_apid,"  APP1.ACC"); /* muß nicht sein */
20:
21:     /* Text für form_alert vorbereiten */
22:     sprintf(name,"[1][Ich, APP1.ACC, habe|die Nummer %d!]
          [ OK ]",gl_apid);
23:     form_alert(1,name);    /* ausgeben */
24:
25:     poi= (long*) &msg[3];  /* Zeiger auf Element 3 */
26:     while(1)               /* bis auf ewig */
27:     {
28:        evnt_mesag(msg);    /* Warten auf... */
29:        if (msg[0]!=111)    /* Nicht unsere besondere */
                               /* Botschaft */
30:           printf("Typ: %d  Sender-ID: %d  2:%d 3:%d 4:%d 5:%d
                6:%d 7:%d\n",msg[0],msg[1],msg[3],msg[4],msg[5],
                msg[6],msg[7]);
31:        else
32:        {
33:           strcpy(name,*poi); /* Kopiere String */
34:           printf("Accessory:: %s",name); /* String anzeigen */
35:        }
36:        gemdos(7);             /* auf Taste warten */
37:        form_dial(3,0,0,640,400,0,0,640,400); /* Bildschirm */
                                                 /* aufbereiten */
38:     }
39:  }
```

Zu Ihren Diensten ...

Wir bieten Ihnen für Ihren ATARI ST eine Reihe leistungsfähiger Dienstprogramme, die Ihnen täglich einen produktiven Einsatz Ihres Rechners ermöglichen.

G COPY — NEU!

Für Vielkopierer, Sicherheitskopierer, Raubkopierer, Umkopierer und nicht für Fotokopierer. Das universelle Kopierprogramm für jedes Programm und für Jedermann.
- formatiert alle Formate (80 - 83 Tracs, 9 - 11 Sektoren, 912 KB Disketten, Schnelladedisketten) in Hochgeschwindigkeit
- kopiert alle ATARI ST Programme und jeden Kopierschutz
- optimale Kopiergeschwindigkeit

DM 99,—

FAST SPEEDER — NEU!

Für alle, die ihrem Massenspeicher keine Pause gönnen wollen. Bringen Sie Festplatte und Diskettenstation auf Trab! Im Extremfall erreichen Sie Ramdiskgeschwindigkeit!
- Festplatten bis zu 200% schneller, Disketten bis zu 900%
- einfachste Handhabung
- optimiertes Cache Prinzip mit Spezialalgorithmus

DM 129,—

G RAMDISK II



neue Version

Warum eine Ramdisk kaufen, wo es viele PD Ramdisks gibt? 1. Die neuen Features überzeugen eindeutig. 2. Wir bieten fachliche Hilfe bei Problemen. 3. einfachste Handhabung. 4. Unsere Ramdisk ist mehrere tausend Mal verkauft und hat sich im harten Einsatz bestens bewährt.
- bis 4 MB Größe
- resetfest und abschaltbar
- Bootcopy
- mit Druckerspooler

DM 49,—

INTERPRINT II



neue Version

Ist bereits bestens bewährt, und was sich problemlos bewährt hat, ist gut. Die universelle Anpassung für jeden Drucker ist ein echter Treiber, der jetzt auch Umlaute konvertiert.
- integrierter Spooler
- integrierte G RAMDISK II
- konvertiert jedes Zeichen
- Hardcopies für alle Matrixdrucker und alle Laserdrucker
- komfortabler Editor
- Parameter speicherbar

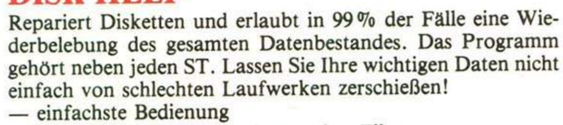
DM 99,—
ohne Ramdisk DM 49,—

Harddisk Help & extension

Wir wollen keinesfalls die Ängste um Ihre täglichen Daten schüren, aber sind Ihre Daten wirklich sicher? Wie schnell führt ein Headcrash zum Ende! Ein Backupsystem mit deutscher Gründlichkeit für eine sicher Datensicherung:
- Partition Backup
- Tree Funktion
- Diskoptimizer
- extrem schnell (s. Test ST 5/87)
- mixed Backup

DM 129,—

DISK HELP



Repariert Disketten und erlaubt in 99% der Fälle eine Wiederbelebung des gesamten Datenbestandes. Das Programm gehört neben jeden ST. Lassen Sie Ihre wichtigen Daten nicht einfach von schlechten Laufwerken zerschießen!
- einfachste Bedienung
- korrigiert Lesefehler, rekonstruiert Files
- ausgenommen physikalische Schäden. Der Kaffee wird nicht von der Diskette entfernt.

DM 79,—

G DISKMON II



Zum Kontrollieren, Ausprobieren, Umkopieren, Raubkopieren, Manipulieren, Interessieren und Reparieren. Ein nützliches und wichtiges Tool, damit die Diskette kein Geheimnis bleibt.
- unbeschränktes Lesen, Schreiben, Editieren, Kopieren, Suchen, Ersetzen, Formatieren usw.
- Bootsektorscannung in Klartextdarstellung
- Schnelldruck
- interaktive Bedienung in Hex, Dez, ASC II

DM 99,—

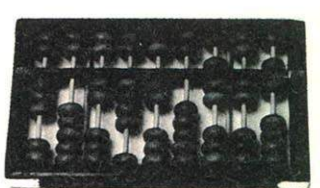
Update Aktion:

G RAMDISK II oder INTERPRINT ohne Ramdisk
—— INTERPRINT mit Ramdisk DM 50,—
AS SOUND SAMPLER
—— AS SOUND SAMPLER II DM 70,—

Alle anderen Produkte erhalten Sie in der jeweils neuesten Version gegen DM 20,—.
Updates werden gegen Einsendung der Originaldiskette und der Bearbeitungsgebühr erledigt.

VIELSEITIG...

...ist unser neues Datenbanksystem geworden. Eine Datenbank mit Maskengenerator, einfacher effektiver Benutzerführung, mit Window und Menütechnik, Help Funktionen, Druckmaskengenerator und Listengenerator?
Selbstverständlich!
Darüberhinaus haben wir unserer neuen Datenbank einige sinnvolle, aber längst nicht selbstverständliche Features mit auf den Weg gegeben.
So werden Sie in **G DATEI** zusätzlich einen Taschenrechner finden.
Rechnen kann unsere Datenbank natürlich auch innerhalb der Datei über Rechenfelder.
Außerdem wurde eine kleine Textverarbeitung integriert, so daß für Serienbriefe kein Springen von Textverarbeitung zur Dateiverwaltung mehr nötig ist.
Wo wir einmal dabei waren, haben wir in **G DATEI** gleich noch ein Expertensystem integriert, das es ermöglicht, die Daten logisch zu verknüpfen oder Entscheidungshilfen anzufordern.





Natürlich ist unser Expertensystem einfach und logisch aufgebaut und ohne Programmierkenntnisse zu bedienen. Sie sehen, wir haben sehr viel Wert darauf gelegt, daß Sie Ihre Daten nicht nur ein-und ausgeben, sondern sie so bearbeiten und auswerten können, wie Sie es wollen.
Einige Leistungsmerkmale:
- Feldtypen Alpha, Numerisch, Datum, Zeit
- 2250 Zeichen pro Datensatz
- Extended Card
- sehr umfangreiche Selektionsmöglichkeiten
- Bildschirmmaskeneditor
- Druckmaskeneditor
- Rechenfelder
- einseitige Serienbriefe
- Druckeranpassung
- Listengenerator mit Überschriften, Summenfeldern, Steuerzeichen

G DATEI kostet trotz all dieser Leistungsmerkmale nur **DM 199,—**

G DATEI ist von der Gesellschaft der unabhängigen EDV-Berater empfohlen!

NEU: GPD Größte Public Domain Sammlung Deutschlands
280 Disketten mit *vollen* 408 KB: Über 5000 Programme

Alle Neuheiten aus USA und GB
Nur bei uns: Disk-Zeit, die aktuelle ST-Diskettenzeitung als PD

PD-Liste gegen DM 2,— Rückporto und Adressaufkleber

G DATA

TELEFONISCHE BESTELLUNGEN 023 25 / 6 08 97



Siemensstraße 16 · 4630 Bochum 1

Schweiz:
Senn Computer AG
Langstraße 31 · CH-8021 Zürich

Österreich
Computershop Rittner
Hauptstr. 34 · A-7000 Eisenstadt

NEU: G DATA BENELUX · Postbus 70 · NL-2000 AB Haarlem Tel. 023 / 32 13 31

Alle angegebenen Preise sind unverbindliche Preisempfehlungen.

Bestell-Coupon

Bitte senden Sie mir:

zzgl. DM 5,— Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort

Algorithmen & Datenstrukturen in PASCAL

Teil 3: Listen

Nachdem wir uns in den beiden ersten Folgen von Algorithmen & Datenstrukturen mit zwei Strukturen zur Datenanordnung (Stacks und Queues) beschäftigt haben, möchte ich heute mit der Datenverarbeitung beginnen.

Geplant sind hier insgesamt drei Unterkapitel, in denen ich mich von den heute behandelten Listen über allgemeine Binär-Bäume zu den AVL-Bäumen vorarbeiten werde.

Motivation

Da die Datenerfassung und Verarbeitung unbestritten eine der Hauptaufgaben des Computers ist, stellt sich die Gretchenfrage:

Wie sollen Daten erfaßt und verarbeitet werden ?

Dazu gibt es, wie sich aus der Einleitung ersehen läßt, viele Ansätze, die aber immer zwei Gemeinsamkeiten haben:

1. Zunächst wird für alle diesen Problemlösungen eine Struktur vereinbart (Liste, Baum, ...).

2. Nach der Festlegung der Struktur müssen hier Daten in einer gewissen Abfolge eingeordnet werden, wofür ein Kriterium (Schlüssel) gefunden werden muß.

Schlüssel

Mit diesen Schlüsseln möchte ich mich zunächst etwas näher befassen.
Die einzige Forderung an Schlüssel ist, anordenbar zu sein, das heißt, daß eine <oder <= – Relation, mittels der eine Einordnung in die Struktur erfolgen kann, existieren muß.

In PASCAL trifft dies auf alle einfachen Datentypen in Zusammenhang mit der 'normalen' <bzw. <= – Relation zu:

- integer, real, char
- string (soweit vorhanden)
- boolean (obwohl das etwas arm wirkt : FALSE<TRUE)
- sämtliche Aufzählungs- und Unterbereichstypen

Natürlich ist es auch möglich, sich selbst auf anderen Datentypen eine Relation zu definieren.

Als Beispiel möchte ich hier Vektoren nennen, die üblicherweise folgendermaßen dargestellt werden:

```
CONST dim = 10;

TYPE vektor-type = ARRAY
1..dim OF real;

VAR vektor : vektor-type;
```

Hieraus kann man einen Schlüssel und somit eine Relation konstruieren, indem man die Länge des Vektors betrachtet :

```
FUNCTION relation(v1, v2:
vektor-type) : boolean;

FUNCTION laenge (v : vektor-
type) : real;
VAR i : integer;
help : real;
BEGIN { laenge }
help:=0;
FOR i:=1 TO dim DO
help:=help+v[i]*v[i];
laenge:=sqrt(help);
END; { laenge }

BEGIN { relation}
relation:=laenge(v1)<=laen-
ge(v2);
END; { relation}
```

(Nichtmathematiker sollte dieses Beispiel nicht abschrecken, es ist für die weitere Betrachtung von keinerlei Bedeutung.)

Angeordnete Listen

Ausgerüstet mit diesem Schlüsselbegriff kann man nun die Anforderungen an unsere Liste wie folgt definieren:

– Zunächst zerfallen sämtliche zu erfassenden Daten in einen Schlüssel (key) und in restliche Daten (data). (Beispiel : Adressen zerfallen in Nachnamen (key) und Restadressen (data)).

– Gemäß diesem Schlüssel wird ein Datensatz (key & data) so in die Liste eingeordnet, daß sämtliche Datensätze der Vorgänger <= dem neuen Datensatz sind, und umgekehrt sämtliche Nachfolger >= dem Datensatz sind.

– Um diese Struktur zu erhalten, werden folgende Operationen benötigt:

1. **create** : Zum Initialisieren einer Liste.
2. **insert** : Zum Einfügen eines Elementes in eine bestehende Liste.
3. **delete** : Zum Löschen eines durch einen Schlüssel bezeichneten Elementes.

Weiterhin muß man Daten natürlich wiederfinden können, dazu die beiden Operatorenpaare:

4a. **find-first** : Liefert das erste Listenelement mit bezeichnendem Schlüssel.

4b. **find-next** : Liefert das jeweils nächste Element.

5a. **first** : Liefert das generell erste Listenelement.

5b. **next** : Liefert das jeweils nächste Element.

Und 'last but not least': Was nutzt uns das Ganze, wenn wir es nicht speichern können? :

6. **save** : Speichert eine komplette Liste.

7. **load** : Lädt eine komplette Liste.

Zeigercharakter

Wie schon zweimal werde ich auch hier zur Realisierung wieder die dynamische Speicherverwaltung bemühen. Betrachten Sie hierzu bitte Listing 3a:
Ein Listenelement **(list-element)** besteht nun, wie man sieht, zunächst aus einem Schlüssel **(key** von **key-type)** und Restdaten **(data** von **data-type)**.

key-type und **data-type** sind nun, unter Beachtung der im Punkt Schlüssel genannten Einschränkungen, frei wählbar.
Der Einfachheit halber habe ich beide zu Integer gewählt.

Der dritte Bestandteil eines Listenelementes ist nun ein Zeiger auf das nachfolgende Listenelement. Die etwas verwirrende Typbezeichnung **list** für den Listennachfolger liegt darin begründet, daß der Listenkopf, also der Zeiger, der den Anfang der Liste markiert (und somit synonym für die komplette Liste steht), ebenfalls nur ein Zeiger auf **list-element** ist.

Als letztes benötige ich noch einen Typ für den Datentransfer zwischen der Liste und der Datei, in der die Daten zu speichern sind **(file-type)**. Da man mit dem Zeiger in der Datei nichts anfangen kann, besteht dieser Dateidatentyp nur aus **key-type** und **data-type**.

Mit diesen Datentypen realisieren sich nun unsere neun Operationen (Listing 3b) folgendermaßen:

1. create(X)

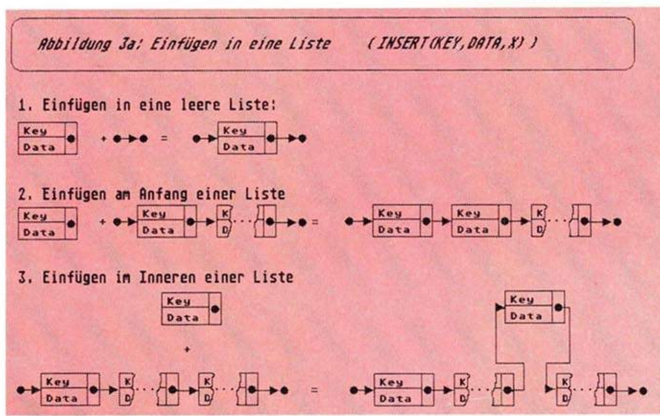
Hier ist nichts weiter zu tun, als die Liste zu nil zu initialisieren.

2. insert(new-key,-new-data,x)

Das Einfügen in eine Liste gestaltet sich in drei Schritten:

- Erzeugung des nötigen Speichers (new).
- Initialisierung des neuen Elementes **(key, data)**.
- Das Einhängen in die Liste, was den kompliziertesten der drei Vorgänge darstellt.

Hier sind drei Fälle zu unterscheiden (Abb. 3a):



Zunächst besteht die Möglichkeit, daß die Liste noch leer ist (x=nil). In diesem Fall ist das einzufügende Element selbst die Liste und der Nachfolger dieses Elementes wird zu nil. Bei nicht leeren Listen (else-Zweig) ist nun wieder die Unterscheidung zu treffen, ob am Anfang der Liste eingefügt werden soll oder im Inneren (IF key<=x̂.key THEN).

Am Anfang einzufügen ist nun wieder der einfache Fall. Der Nachfolger des neuen Elementes **(next)** ist dann die Liste, die dann mit dem neuen Element **(x:=new-element)** beginnt.

Im anderen Fall ist die Liste zu durchlaufen. Dazu wird ein Hilfszeiger **(work-ptr)** benutzt, der zunächst auf **x** zeigt. Die Liste wird solange durchlaufen, bis entweder das Listenende erreicht ist oder ein passender Schlüssel gefunden

G SCANNER

Unser neuer S/W-Scanner zum problemlosen Digitalisieren von Bildern aller Art.
Die hier gezeigten Bilder sind mit dem G SCANNER gescannt worden. Wir verwenden ausschließlich hochwertige HP-Dioden (Hewlett Packard) als Reflexgeber. Die Diode wird einfach auf dem Druckkopf Ihres Matrixdruckers befestigt. Hierfür ist jeder Matrixdrucker geeignet und kann per Software angepaßt werden.



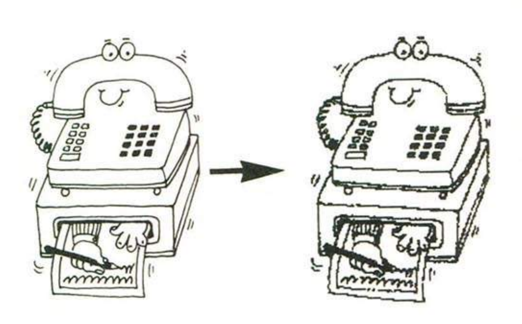
Die Vorlage wird in den Drucker eingespannt und erscheint während des Digitalisierungsvorganges auf dem Bildschirm.
Der G SCANNER erfordert keinerlei Lötarbeiten, sondern wird einfach in den ROM Port des ATARI gesteckt.



Die erreichbare Auflösung beträgt volle 200 DPI.
Mitgeliefert wird Software zum Scannen, die DEGAS und Normalbilder erstellt und einfaches Nachbearbeiten ermöglicht.
Der G SCANNER ist voll kompatibel zu STAD. Die Scannoption im Programm kann damit endlich voll genutzt werden.

Zubehör: NEC P6 Papierhalterung für Diode: DM 35,—

DM 298,—



Original **G SCANN't**

TV MODULATOREN

Unsere Modulatoren ermöglichen endlich den Anschluß Ihres ST an *jeden* Fernseher. Dieser wird dabei als Farbmonitor benutzt.
Auf Farbfernsehern werden nicht nur *alle* Farben des ST dargestellt, sondern auch sämtliche Mischtöne, die der ST erzeugen kann.
Auch der Sound kommt nicht zu kurz. Wahlweise kann der TV-Lautsprecher oder eine Stereoanlage zur Tonausgabe benutzt werden. Dabei ist selbstverständlich ein Extra-Anschluß für Stereoanlagen eingebaut.
Das heißt: bestes Bild und bester Ton an jedem Fernseher.
Die Modulatoren werden anschlußfertig mit separatem Netzteil geliefert und können problemlos an alle Fernseher und alle STs angeschlossen werden.

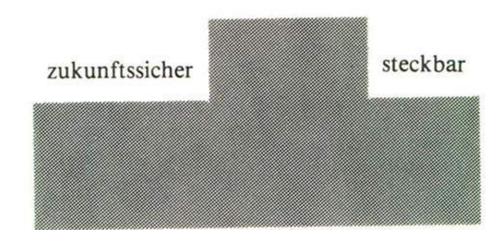
Wir liefern 3 Versionen. Typ A hat einen zusätzlichen Monitorumschalter, so daß Sie ohne Steckerwechsel zwischen TV (Farbe) und dem Original-Monochrommonitor umschalten können.
Typ B wird ohne Umschalter geliefert.
Typ C ist ein FBAS Wandler, mit dem sich Fernseher und Videorecorder mit Video (AV) Eingang oder FBAS Monitore anschließen lassen. Ohne Monitorumschalter und Antennensignal.

Typ A mit Monitorumschalter: DM 249,—
Typ B ohne Monitorumschalter: DM 239,—
Typ C FBAS Wandler: DM 169,—

RAM Erweiterungen

Umsere RAM Erweiterungen stellen ein völliges Novum auf dem Erweiterungsbereich dar.
1. Sie werden ohne Lötarbeiten eingebaut, es sind weder Umlötungen von Leiterbahnen noch irgendwelche Treiberprogramme erforderlich. Lassen Sie sich keinen Bären aufbinden: Gesteckte Erweiterungen sind für jedermann problemlos einzubauen und mindestens so zuverlässig wie gelötete.

Nur *eine* Erweiterung
Für 1, 2 oder 4MB



2. Unsere Erweiterungsplatine wird unbestückt geliefert. Sie läßt sich – je nach Bedarf und Geldbeutel – mit 256 KBit oder MBit Chips bestücken und erweitert so den RAM-Bereich um 512KB, 2MB oder 4MB. Auf diese Art kann man zunächst um 512KB erweitern, später problemlos auf 4MB. Alles mit einer Platine.
Aktuelle Preise von 256KBit und 1MBit auf Anfrage.

Platine für 520, 520+, 260, 520 STM DM 249,—
Platine für 1040 STF DM 249,—

NEU: GPD Größte Public Domain Sammlung Deutschlands
280 Disketten mit *vollen* 408 KB: Über 5000 Programme
Alle Neuheiten aus USA und GB
Nur bei uns: Disk-Zeit, die aktuelle ST-Diskettenzeitung als PD
PD-Liste gegen DM 2,— Rückporto und Adressaufkleber

G DATA

TELEFONISCHE BESTELLUNGEN 023 25 / 6 03 97



Siemensstraße 16 · 4630 Bochum 1

Schweiz:
Senn Computer AG
Langstraße 31 · CH-8021 Zürich

Österreich
Computershop Rittner
Hauptstr. 34 · A-7000 Eisenstadt

NEU: G DATA BENELUX · Postbus 70 · NL-2000 AB Haarlem · Tel. 023 / 32 13 31

Alle angegebenen Preise sind unverbindliche Preisempfehlungen.

Bestell-Coupon

Bitte senden Sie mir:

zzgl. DM 5,— Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

wurde. Wichtig hierbei ist, daß man vorausschauend abprüft, also immer den Schlüssel des Nachfolgers **(work.next.-key)** betrachtet.

Um diese Notwendigkeit einzusehen, betrachten Sie bitte den dritten Fall der Abbildung 3a.

Wie Sie sehen, müssen zum Einfügen eines Elementes in eine Liste
sowohl der Zeiger **work-ptr.-next** als auch der **next**-Zeiger des neuen Elementes umgesetzt werden. Würde die Liste nicht vorausschauend durchlaufen, könnte man **work-ptr.next** nicht mehr umsetzen, weil man damit schon beim nächsten Element angelangt wäre.

Wenn bis zum Ende der Liste noch nicht eingefügt wurde (IF NOT fertig), ist das neue Element das letzte Listenelement, und die Zeiger müssen entsprechend umgesetzt werden.

3. delete(key,x)

Bei der Löschoperation **delete(key,x)** müssen zwei Fälle unterschieden werden:

Beim Löschen am Anfang der Liste muß durch Umsetzen des Listenzeigers auf seinen Nachfolger das erste Element herausgenommen werden.

Beim Löschen innerhalb der Liste ergibt sich nun wieder, wie schon bei **insert**, die Notwendigkeit, die Liste vorausschauend zu durchlaufen. Deswegen hat die hierzu notwendige Konstruktion (WHILE ... DO) eine gewisse Ähnlichkeit mit der bei **insert.** (Sie unterscheiden sich nur in den Zeigerumsetzungen.)

Abbildung 3b: Löschen eines Listenelementes (DELETE(KEY,X))

1. Löschen am Anfang der Liste

2. Löschen innerhalb der Liste

3. Löschen des letzten Elementes

Key / Data

Da bei **delete** die Möglichkeit besteht, daß Herausnahmevorgang fehlschlägt (kein Element mit Schlüssel **key**), muß dessen Erfolg oder Mißerfolg dem aufrufendem Programm mitgeteilt werden. Aus diesem Grund ist **delete** eine Funktion mit Ergebnisparameter boolean.

Ein weiteres Problem von **delete** besteht darin, daß bei den bisherigen Definitionen davon ausgegangen wurde, daß auch doppelte Schlüsselbelegungen, also beispielsweise zweimal der Schlüssel >1<, möglich sind.

Da bei **delete** aber nur der jeweils erste Eintrag mit Schlüssel **key** gelöscht wird, ist es erforderlich, durch wiederholten Aufruf von **delete** sämtliche Einträge zu entfernen, wenn dies erwünscht ist.

Die erforderliche PROCEDURE ist im Modul selbst nicht formuliert, könnte aber folgendermaßen aussehen:

```
PROCEDURE delete-keys(
key : key-type;
VAR x : list);
BEGIN {delete-keys}
WHILE delete(key,x) DO ;
END; {delete-keys}
```

4. find-first(x,key,-data) und find-next(key,data)

Um nun Informationen aus einer Liste **x** zu bekommen, muß es möglich sein, in dieser Liste nach bestimmten Schlüsseln zu suchen.

Diese Aufgabe kommt dem Funktionspaar **find-first** und **find-next** zu.

Seine Arbeitsweise hat man sich so vorzustellen, daß die Funktion **find-first** den ersten Datensatz der Liste mit Schlüssel key liefert und **find-next** nacheinander sämtliche weiteren Datensätze berechnet.

Benutzt wird hierzu eine Hilfsfunktion **find** und eine globale Variable **akt-find.**

find hat nun die Aufgabe, die Liste ab der Position **akt-find** zu durchsuchen. Dazu wird die Liste mit dem Arbeitszeiger **akt-find** durchlaufen (WHILE ...), bis das Ende erreicht ist oder ein **key** entsprechender Datensatz gefunden wurde.

Bei **find-first** ist zu beachten, daß vor dem Aufruf von **find** der Zeiger **akt-find** mit dem Listenkopf initialisiert wird.

find-next schließlich begnügt sich mit dem Aufruf von find.

Bei beiden Funktionen ist zu beachten, daß der Funktionswert jeweils die Gültigkeit des Datensatzes angibt.

Dies bedeutet, beide Funktionen liefern einen gültigen Wert, wenn der Funktionsaufruf **true** ergibt, sonst einen undefinierten Wert.

5. first(x,key,data) und next(key,data)

Bei diesen beiden Funktionen handelt es sich um eine abgemagerte Variante der vorherigen Funktionen.

Es wird nicht nach einem bestimmten Schlüssel gesucht, sondern das jeweils nächste Listenelement einer Liste **x komplett** (Schlüssel & Restdaten) ausgegeben.

Dazu wird ebenfalls wieder eine globale Variable **akt-list** benötigt, die die bis jetzt erreichte Stelle in der Liste **x** markiert. Die Arbeitsweise ist klar:

Bei **first wird zunächst ein Check auf eine gefüllte Liste durchgeführt (gefunden:= x<>nil)**. Liegt diese vor (IF ... THEN), kann man die Rückgabewerte belegen **(key, data).** Der Zeiger **akt-list** bekommt dabei den Wert von **x,** da ja gerade das erste Element untersucht wurde. Der FUNCTION **first** muß nun nur noch der Rückgabeparameter **gefunden** mitgeteilt werden.

next schließlich funktioniert völlig analog, nur daß man hier nicht die Liste, die beim Zeiger **x** beginnt, betrachtet, sondern die Liste, die beim Zeiger **akt-list** beginnt, also die Restliste.

Die Problematik dieser beiden Funktionen, wie schon der Funktionen unter 4, ist die Tatsache, daß globale Variablen benutzt werden. Wenn man nun, wie ich, die FUNCTIONen/PROCEDUREn als Header-File vereinbart (siehe auch Testumgebung, Listing 3c), ist es, wegen der etwas restriktiven PASCAL-Syntax, erforderlich, das Modul der Listenoperationen (Listing 3b) zuerst zu benennen.

Begründung : In PASCAL dürfen Variablen und Funktions-/Procedure-Deklarationen nicht gemischt werden (anders als z.B. in C).

6. save(x,filename)

Da die Bedeutung dieses Befehls wohl klar sein dürfte, komme ich gleich zur Programmierung:

Was Sie hier sehen, ist eine typische Anwendung der beiden vorhergehenden Funktionen **first** und **next.**

Nach der schreibenden Öffnung der durch **filename** bezeichneten Datei werden zunächst mit **first,** später mit **next,** die Schlüssel und die Restdatendefinitionen in den Dateipuffer **t** geschrieben und später mit **put(t)** in die Datei **filename** selbst befördert.

7. load(x,filename)

Auch beim Laden von Listen kann man eine Anwendung von zwei anderen Listenoperationen wiederfinden, nämlich die Anwendung von **create(X)** und **insert(key,data,x).**

Ähnlich wie bei save wird hier zunächst die Datei mit Namen **filename** eröffnet, diesmal allerdings mit lesendem Zugriff.

Nach der Initialisierung der zu ladenden Liste **(create(x))** erfolgt deren Aufbau mittels **insert(key,data,x).**

Dazu wird die Pufferinformation der Datei **(key & data)** in die Liste **x** eingefügt.

Testumgebung

Nachdem ich oben, wie ich hoffe in ausreichendem Umfang, auf die Programmierung von Listen eingegangen bin, möchte ich nun auf die Anwendung der Listenoperationen zu sprechen kommen.

Dazu habe ich eine kleine Testumgebung für die Module geschrieben (Listing 3c).

Hier wird eine Liste mit den oben beschriebenen Operationen verwaltet, das heißt, es ist möglich eine Liste auf- und abzubauen, sie nach bestimmten Kriterien zu durchsuchen, sie zu speichern und wieder zu laden.

Bei der Programmierung ist zunächst der Deklarationsteil interessant. Man sieht als erstes den Aufruf der Typvereinbarungen durch $I.

Die Variablenvereinbarung beinhaltet hauptsächlich die Deklaration der Liste **(x** : list;) sowie die Deklaration zweier Hilfsvariablen **schluessel** und **daten.**

Mit der Variablen **frage** wird die Benutzereingabe vorgenommen, und in **str** kann ein Dateiname eingelesen werden.

Auf die Variablendeklaration erfolgt sofort (siehe auch Anmerkung unter 5.) der Aufruf des Moduls list.pas (Dieses enthält unsere Listenoperationen und die beiden globalen Variablen).

Im Anweisungsteil erfolgt nun

nichts Erstaunliches mehr.

Nach der Initialisierung unserer Schlange **x** mit **create** wird eine REPEAT-Schleife zwecks Wiederholung des Benutzerdialoges betreten.

Nach der Ausgabe eines Kommandomenüs und der Wahl einer entsprechenden Funktion erfolgt die Abarbeitung der Funktion ganz im Sinne der oben definierten Listenoperationen.

Bemerkenswert ist hier vielleicht noch das Zusammenwirken von **first** und **next** bzw. **find–first** und 2 find–next bei dem Auflisten sämtlicher Elemente einer Liste bzw. dem Auflisten sämtlicher Elemente mit gleichem Schlüssel (Menüpunkte <3>und <4>).

Listing 3 a

```
 1:   { Listing 3a)
 2:
 3:     Dieses Modul stellt die Typen fuer den Datentyp LIST zur
          Verfuegung.
 4:
 5:     Reservierte Woerter : list, key_type, data_type,
          list_element, file_type
 6:
 7:     programmiert          : Mai 87
 8:     von                   : Dirk Brockhaus
 9:     mit                   : PASCAL+ (CCD)
10:   }
11:
12:   list          = ^list_element;
13:
14:   key_type      = integer;
15:
16:   data_type     = integer;
17:
18:   list_element = RECORD
19:                    key  : key_type;
20:                    data : data_type;
21:                    next : list;
22:                  END;
23:
24:   file_type     = RECORD
25:                    key  : key_type;
26:                    data : data_type;
27:                  END;
28:
```

Listing 3 b

```
 1:   { Listing 3b)
 2:
 3:     Dieses Modul stellt die Operationen auf dem
          Datentyp LIST zur Verfuegung.
 4:
 5:     Funktionalitaet der Operationen:
 6:
 7:      - create: () --> LIST
 8:      - insert: key x data x list --> list
 9:      - delete: key x list --> list x boolean
10:      - find_first : list x key --> data x
           boolean
11:      - find_next  : list x key --> data x
           boolean
12:      - first  : list --> key x data x boolean
13:      - next : list --> key x data x boolean
14:      - save : list x string --> file
15:      - load : file x string --> list
16:
17:     Weiterhin besitzt das Modul eine lokal be-
          noetigte Operation :
18:
19:      - find : list x key --> data x boolean
20:
21:     Sowie zwei lokale Variablen : akt_list,
          akt_find
22:
23:     Reservierte Woerter : create, insert,
          delete, find_first, find_next,
24:       first, next, find, load, save, akt_list,
          akt_find
25:
26:     programmiert          : Mai 87
27:     von          : Dirk Brockhaus
28:     mit          : PASCAL+ (CCD)
29:   }
30:
31:   akt_list ,
32:   akt_find : list;
33:
34:   PROCEDURE create (VAR x : list);
35:
36:     BEGIN {create}
37:       x:=nil;
38:     END: {create}
39:
40:   PROCEDURE insert (     new_key   : key_type;
41:              new_data : data_type;
42:            VAR x          : list);
43:
44:     VAR work_ptr     ,
45:         new_element : list;
46:         fertig     : boolean;
47:
48:     BEGIN {insert}
49:       new(new_element);
50:       WITH new_element^ DO
51:         BEGIN
52:     key:=new_key;
53:     data:=new_data;
54:     IF x=nil THEN
55:       BEGIN
56:         next:=nil;
57:         x:=new_element;
58:       END
59:     ELSE
60:       IF key<=x^.key THEN
61:         BEGIN
62:           next:=x;
63:           x:=new_element;
64:         END
65:       ELSE
66:         BEGIN
67:           work_ptr:=X;
68:           fertig:=FALSE;
69:           WHILE (work_ptr^.next<>nil) AND NOT
                fertig DO
70:       IF work_ptr^.next^.key>key THEN
71:         BEGIN
72:           next:=work_ptr^.next;
73:           work_ptr^.next:=new_element;
74:           fertig:=true;
75:         END
76:       ELSE
77:         work_ptr:=work_ptr^.next;
78:           IF NOT fertig THEN
79:       BEGIN
80:         next:=work_ptr^.next;
81:         work_ptr^.next:=new_element;
82:       END;
83:         END;
84:         END;
85:     END; {insert}
86:
87:   FUNCTION delete(     key : key_type;
88:       VAR x : list) : boolean;
89:
90:     VAR work_ptr : list;
91:         gefunden : boolean;
92:
93:     BEGIN {delete}
94:       gefunden:=false;
95:       IF x<>nil THEN
96:         BEGIN
97:     IF x^.key=key THEN
98:       BEGIN
99:         x:=x^.next;
100:        gefunden:=true;
101:      END
102:    ELSE
103:      BEGIN
104:        work_ptr:=x;
105:        WHILE (work_ptr^.next<>nil) AND NOT
              gefunden DO
106:          IF work_ptr^.next^.key=key THEN
107:        BEGIN
108:        work_ptr^.next:=work_ptr^.next^.next;
109:        gefunden:=true;
110:      END
111:          ELSE
112:      work_ptr:=work_ptr^.next;
113:      END;
114:        END;
115:      delete:=gefunden;
116:    END: {delete}
117:
118:  FUNCTION find(     key   : key_type;
119:          VAR data : data_type) : boolean;
120:
121:    VAR gefunden : boolean;
122:
123:    BEGIN {find}
124:      gefunden:=false;
125:      WHILE (akt_find<>nil) AND NOT gefunden DO

126:        BEGIN
127:    IF akt_find^.key=key THEN
128:      BEGIN
129:        gefunden:=true;
130:        data:=akt_find^.data;
131:      END;
132:    akt_find:=akt_find^.next;
133:        END;
134:      find:=gefunden;
135:    END; {find}
136:
137:  FUNCTION find_first(  x     : list;
138:        key  : key_type;
139:          VAR data : data_type) : boolean;
140:
141:    BEGIN {find_first}
142:      akt_find:=x;
143:      find_first:=find(key,data);
144:    END; {find_first}
145:
146:  FUNCTION find_next(     key   : key_type;
147:        VAR data : data_type) : boolean;
148:
149:    BEGIN {find_next}
150:      find_next:=find(key,data);
151:    END; {find_next}
152:
```

Listing 3 b
Fortsetzung

```
153:  FUNCTION first(    x  : list;
154:            VAR key  : key_type;
155:            VAR data : data_type) : boolean;
156:
157:    VAR gefunden : boolean;
158:
159:    BEGIN {first}
160:      gefunden:=x<>nil;
161:      IF gefunden THEN
162:        BEGIN
163:    akt_list:=x;
164:    key:=x^.key;
165:    data:=x^.data;
166:        END;
167:      first:=gefunden;
168:    END; {first}
169:
170:  FUNCTION next(VAR key  : key_type;
171:           VAR data : data_type) : boolean;
172:
173:    VAR gefunden : boolean;
174:
175:    BEGIN {next}
176:      gefunden:=akt_list^.next<>nil;
177:      IF gefunden THEN
178:        BEGIN
179:    akt_list:=akt_list^.next;
180:    key:=akt_list^.key;
181:    data:=akt_list^.data;
182:        END;
183:      next:=gefunden;
184:    END; {next}
185:
186:  PROCEDURE save(x  : list;
187:            filename : string);
188:
189:    VAR t : FILE OF file_type;
190:
191:    BEGIN {save}
192:      rewrite(t,filename);
193:      IF first(x,t^.key,t^.data) THEN
194:        BEGIN
195:    put(t);
196:    WHILE next(t^.key,t^.data) DO
197:      put(t);
198:        END;
199:    END; {save}
200:
201:  PROCEDURE load(VAR x      : list;
202:           filename : string);
203:
204:    VAR t : FILE OF file_type;
205:
206:    BEGIN {load}
207:      reset(t,filename);
208:      create(x);
209:      WHILE NOT eof(t) DO
210:        BEGIN
211:    insert(t^.key,t^.data,x);
212:    get(t);
213:        END;
214:    END; {load}
215:
```

Listing 3 c

```
 1:  { Listing 3c)
 2:
 3:    Testumgebung zu den LIST-Modulen.
 4:    Es wird eine Liste verwaltet, in der eingefuegt und ge-
         loescht werden
 5:    kann. Weiterhin ist eine Komplettausgabe moeglich, sowie
         die Ausgabe
 6:    von Daten zu bestimmten Schluesseln. Die Liste kann auch
         auf Diskette
 7:    abgelegt und spaeter wieder eingeladen werden.
 8:
 9:    programmiert : Mai 87
10:    von          : Dirk Brockhaus
11:    mit          : PASCAL+ (CCD)
12:  }
13:
14:  PROGRAM teste_listen_operationen (input,output);
15:
16:    TYPE {$I list_typ.pas}
17:
18:    VAR x    : list;
19:        schluessel : key_type;
20:        daten  : data_type;
21:        frage  : char;
22:        str   : string;
23:
24:    {$I list.pas}
25:
26:    BEGIN
27:      writeln('Testumgebung zum Modul LIST:');
28:      writeln;
29:      create(x);
30:      REPEAT
31:        writeln('<1> Fuege Element ein');
32:        writeln('<2> Loesche Element');
33:        writeln('<3> Finde bestimmte Elemente');
34:        writeln('<4> Liste alle Elemente');
35:        writeln('<5> Speichere Liste');
36:        writeln('<6> Lade Liste');
37:        writeln('<0> Beenden');
38:        writeln;
39:        write('? ');
40:        read(frage);
41:        writeln;
42:        CASE frage OF
43:      '1' : BEGIN
44:       writeln('Element einfuegen !');
45:       write('Schluessel ? ');
46:       readln(schluessel);
47:       write('Daten ? ');
48:       readln(daten);
49:       insert(schluessel,daten,x);
50:           END;
51:      '2' : BEGIN
52:       writeln('Element loeschen !');
53:       write('Schluessel ? ');
54:       readln(schluessel);
55:       IF NOT delete(schluessel,x) THEN
56:         writeln('Nicht gefunden !');
57:           END;
58:      '3' : BEGIN
59:       writeln('Liste Elemente mit folgendem Schluessel !');
60:       write('Schluessel ? ');
61:       readln(schluessel);
62:       IF find_first(x,schluessel,daten) THEN
63:         BEGIN
64:           writeln('Daten : ',daten);
65:           WHILE find_next(schluessel,daten) DO
66:             writeln('Daten : ',daten);
67:         END
68:       ELSE
69:         writeln('Keine Elemente mit diesem Schluessel !');
70:           END;
71:      '4' : BEGIN
72:       writeln('Liste aller Elemente !');
73:       IF first(x,schluessel,daten) THEN
74:         BEGIN
75:           writeln('Schluessel : ',schluessel,' Daten : ',daten
             );
76:           WHILE next(schluessel,daten) DO
77:             writeln('Schluessel : ',schluessel,' Daten : ',
                 daten);
78:         END
79:       ELSE
80:         writeln('Leere Liste');
81:           END;
82:      '5' : BEGIN
83:       writeln('Speicherung der Liste !');
84:       write('Dateiname ? ');
85:       readln(str);
86:       save(x,str);
87:           END;
88:      '6' : BEGIN
89:       writeln('Laden einer Datei !');
90:       write('Dateiname ? ');
91:       readln(str);
92:       load(x,str);
93:           END;
94:         END;
95:       UNTIL frage='0';
96:   END.
```

Vorausschau

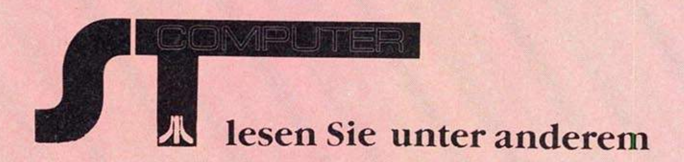
Nach dem glücklichen Abschluß der Listen werde ich mich in der nächsten Folge von Algorithmen & Datenstrukturen auf Binär-Bäume stürzen. Sie stellen eine weitere Struktur zur Datenverarbeitung dar und sind in gewissem Sinne optimaler als Listen, wenn man ihre Einfüge- und Suchzeiten vergleicht. Doch dazu später mehr.

Für heute möchte ich mich bei Ihnen für Ihr Interesse bedanken und erwarte unser Wiedersehen im nächsten Monat.

(Sig)

INSERENTENVERZEICHNIS

In der nächsten



lesen Sie unter anderem

Digitale Speicher

Wer kennt sie nicht? Die kleinen, elektronischen Tausendfüßler, die unser Computerleben so bestimmen, sind heutzutage in nahezu jedem Gerät zu finden. Wir wollen Ihnen einen kleinen Überblick über diese digitalen Speicher geben, die nur Einsen und Nullen kennen, und erklären, was sie alles leisten können.

Chessbase - Schach mal anders

In welcher Disziplin sind die bundesdeutsche und die englische Nationalmannschaft zu Trainingszwecken mit Atari ST ausgerüstet? Ganz zu schweigen vom amtierenden Weltmeister! Was bewegt ältere russische Emigranten, sich mit Dropdown-Menüs und Dialogboxen auseinanderzusetzen, und warum schwärmen kubanische Bohémiens von einer Harddisk? Die Antwort heißt "ChessBase" und ist eine hochspezialisierte Datenbank für Schachpartien, die in Bonn entwickelt wurde. Ein Bericht folgt in der Februar-Ausgabe.

EPROM-Brenner als Selbstbauprojekt

Nach dem Easytizer folgt nun das nächste Hardwareprojekt für den schmalen Geldbeutel. Mit dem "Merlin"-Prommer lassen sich alle EPROMs der 27-Serie brennen (2716 bis 27011). Wir liefern Ihnen Schaltbild und Softwaredokumentation, so daß auch Sie z.B. mal schnell eine Änderung am TOS machen können. Es bleiben keine Geheimnisse offen.

CRP-Digitalisiertablett

Im Zeitalter der fortgeschrittenen CAD- und Grafikprogramme, darf auch ein Digitalisiertablett am ATARI ST nicht fehlen. Mit einem solchen Tablett und Griffel lassen sich auf einfache Art und Weise, Grafiken erstellen. Wir testeten das Produkt der Firma CRP. Es wird u.a. von Programmen wie Campus CAD oder STAD unterstützt.

Neues Pro Fortran 77

Mit seiner neuen Version bringt die englische Firma Prospero nun endlich ein Fortran 77 heraus, das auch die GEM-Programmierung voll unterstützt. Was es sonst noch an Neuigkeiten bei diesem immer noch einzigen Fortran auf dem ST gibt, lesen Sie in der nächsten Ausgabe.

Änderungen vorbehalten !

Die ST-Computer Ausgabe 2/88 erscheint am 29.1.1988

Impressum ST-Computer

Verlag:
Heim Fachverlag,
Heidelberger Landstraße 194,
6100 Darmstadt 13,
Telefon (06151) 56057 FAX 06151/55689

Verlagsleitung:
Hans-Jörg Heim

Redaktion:
'Merlin' Computer GmbH
Uwe Bärtels (UB Chefredakteur)
Marcelo Merino (MM)
Harald Egel (HE)
Harald Schneider (HS)

Anschrift:
'Merlin' Computer GmbH
ST-Computer Redaktion
Postfach 5969
Industriestr. 26
6236 Eschborn
Tel. 06196/481811 FAX 06196/41137

Redaktionelle Mitarbeiter:
Claus Brod (CB), Uli Eickmann (UE), Klaus Heuer (KH), Stefan Höhn (SH), Raymund Hofmann (RH), Oliver Joppich (OJO), Jürgen Leonhard (JL), Claus P. Lippert (CPL), Markus Nerding (MN), Chr. Schormann (CS), Andreas Suchy (AS), Jörg Wilhelm (JW)

Kommunikation und Presseleitung:
Claus P. Lippert

Autoren dieser Ausgabe:
D. Brockhaus, H. Dampel, A. Esser, A. Gieriet, H. P. Labude, P. Lerailler, Th.Pfoch, A. Pötz, Dr. K. Sarnow, W. Thomas, F. Wübbeling

Produktion:
Klaus Schultheis (Ltg.), Patricia Illing, Bernd Uwe Failer, Susanne Failer, Karl-Heinz Hoffmann, Bela Kumar, Christina Nungesser, Ilka Lindemann

Grafische Gestaltung:
U. Berndt, U. Fischer, K. Ohlenschläger

Anzeigenverkaufsleitung:
Uwe Heim

Anzeigenpreise:
nach Preisliste Nr. 2, gültig ab 1.7.86

Vertrieb:
Hans-Jörg Heim, Uwe Heim,
Heide Schultheis

Erscheinungsweise:
11 x jährlich

Bezugspreis:
Einzelheft DM 7,–.
Jahresabonnement DM 70,– inklusive der gesetzlichen Mehrwertsteuer und den Zustellgebühren für 11 Ausgaben
Ausland: 90,– DM inkl. Versand

Bezugsmöglichkeiten:
ATARI-Fachhändler, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag unter obiger Adresse.

Druck:
Ferling Druck Darmstadt

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit ihrer Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern im Heim Verlag. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

Titelseite: Fabian & Mayer
Foto: Rainer Spirandelli

ISSN 0932-0385

hardware
software
organisation
service

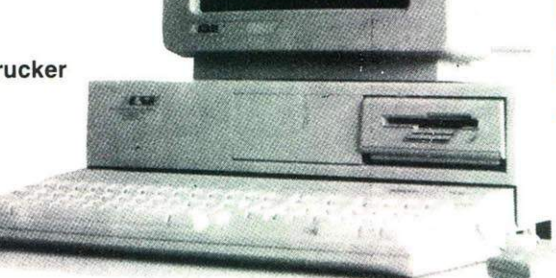
CSF GMBH

Heeper Str. 106–108, 4800 Bielefeld 1, 05 21/6 16 63

Kein Kabelsalat mehr mit dem Gehäuse für ATARI ST

- Zentrale Stromversorgung für alle Geräte einschl. 2 Drucker
- Einbaumöglichkeit von 2 Diskettenlaufwerken
- Rechner (Tastatur) kann komplett unter das Gehäuse geschoben werden (Staubschutz)
- Massives Blechgehäuse

ATARI ST-Gehäuse erhalten Sie bei den autorisierten Fachhändlern



NEU: DM 149,–

NEUES VON CSF

- **Speichererweiterungen auf 2,5 MB**
  steckbar für ATARI 520 ST, 520 ST+, 260 ST, 520 STM 1040 ST
  mit schnellen 1 MB RAM **849,– DM**
- **Speichererweiterungen auf 1 MB,**
  steckbar oder zu Löten
  für ATARI 520 ST, 260 ST, 520 STM **198,– DM**

- **steckbar** (ohne jegliche **Lötarbeiten**)
- läuft auch auf dem 520 STM
- enorme Zeitersparnis durch einfache, bebilderte Einbauanleitung
- kein Flimmern nach der Erweiterung (durch separate, geglättete Spannung an der zweiten RAM-Bank)
- sensationeller Preis
- Bei Bestellungen bitte angeben: ☐ Speichererweiterung steckbar ☐ Speichererweiterung zum Löten

Zu beziehen:	In der Schweiz:	In Österreich:
Direkt bei CSF, Bielefeld **Tel. 05 21/6 16 63** **Bei allen ATARI-Händlern**	**SENN Computer AG** **Langstr. 31 · CH-8021 Zürich** **Tel. 01-241 73 73**	**Institut für Datenverarbeitung und Organ. Ges.mbH** **Rehberger Hauptstr. 95 · A-3503 Krems** **Tel. 0 27 32-7 05 81**

Alle Preise sind unverb. empf. Verkaufspreise



